# Improved Gaussian process modeling : Application to Bayesian optimization

Sébastien Petit

université PARIS-SACLAY

# Improved Gaussian process modeling. Application to Bayesian optimization.

*Améliorations des modèles par processus gaussiens. Application à l'optimisation bayésienne.*

**Thèse de doctorat de l'université Paris-Saclay**

École doctorale n° 580, Sciences et Technologies de l'Information et de la Communication (STIC)
Spécialité de doctorat : Mathématiques et informatique
Graduate School : Informatique et sciences du numérique, Référent : CentraleSupélec

Thèse préparée au Laboratoire des signaux et système (Université Paris-Saclay, CNRS, CentraleSupélec), 91190, Gif-sur-Yvette, France, sous la direction de Emmanuel VAZQUEZ, Professeur des Universités, le co-encadrement de Julien BECT, Maître de Conférence, Jean DEMANGE, Ingénieur-Docteur, Paul FELIOT, Ingénieur-Docteur et Sébastien DA VEIGA, Ingénieur-Docteur.

**Thèse soutenue à Paris-Saclay, le 2 septembre 2022, par**

**Sébastien PETIT**

## Composition du jury

| | |
|---|---|
| **Rodolphe Le Riche**<br>Directeur de Recherche, CNRS, École des Mines de Saint-Etienne | Président |
| **Luc Pronzato**<br>Directeur de Recherche, CNRS, Université Côte d'Azur | Rapporteur & Examinateur |
| **David Ginsbourger**<br>Professeur des Universités, IMSV, Université de Berne | Rapporteur & Examinateur |
| **Amandine Marrel**<br>Ingénieure de Recherche, CEA (DER/SESI) | Examinatrice |
| **Emmanuel Vazquez**<br>Professeur des Universités, CentraleSupélec, Université Paris-Saclay | Directeur de thèse |

THESE DE DOCTORAT
NNT : 2022UPASG063

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

**Titre :** Améliorations des modèles par processus gaussiens. Application à l'optimisation bayésienne.
**Mots clés :** Processus gaussiens, méthodes bayésiennes, choix de modèle, méthodes à noyau, optimisation

**Résumé :** Cette thèse s'inscrit dans la lignée de travaux portant sur la modélisation bayésienne de fonctions par processus gaussiens, pour des applications en conception industrielle s'appuyant sur des simulateurs numériques dont le temps de calcul peut atteindre jusqu'à plusieurs heures. Notre travail se concentre sur le problème de sélection et de validation de modèle et s'articule autour de deux axes.

Le premier consiste à étudier empiriquement les pratiques courantes de modélisation par processus gaussien stationnaire. Plusieurs problèmes sur la sélection automatique de paramètre de processus gaussien sont considérés. Premièrement, une étude empirique des critères de sélection de paramètres constitue le cœur de cet axe de recherche et conclut que, pour améliorer la prédictivité des modèles, le choix d'un critère de sélection parmi les plus courants est un facteur de moindre importance que le choix a priori d'une famille de modèles. Plus spécifiquement, l'étude montre que le paramètre de régularité de la fonction de covariance de Matérn est plus déterminant que le choix d'un critère de vraisemblance ou de validation croisée. De plus, l'analyse des résultats numériques montre que ce paramètre peut-être sélectionné de manière satisfaisante par les critères, ce qui aboutit à une recommandation permettant d'améliorer les pratiques courantes. Ensuite, une attention particulière est réservée à l'optimisation numérique du critère de vraisemblance. Constatant, comme Erickson et al. (2018), des inconsistances importantes entre les différentes librairies disponibles pour la modélisation par processus gaussien, nous proposons une série de recettes numériques élémentaires permettant d'obtenir des gains significatifs tant en termes de vraisemblance que de précision du modèle. Enfin, les formules analytiques pour le calcul de critère de validation croisée sont revisitées sous un angle nouveau et enrichies de formules analogues pour les gradients. Cette dernière contribution permet d'aligner le coût calculatoire d'une classe de critères de validation croisée sur celui de la vraisemblance.

Le second axe de recherche porte sur le développement de méthodes dépassant le cadre des modèles gaussiens stationnaires. Constatant l'absence de méthode *ciblée* dans la littérature, nous proposons une approche permettant d'améliorer la précision d'un modèle sur une plage d'intérêt en sortie. Cette approche consiste à relâcher les contraintes d'interpolation sur une plage de relaxation disjointe de la plage d'intérêt, tout en conservant un coût calculatoire raisonnable. Nous proposons également une approche pour la sélection automatique de la plage de relaxation en fonction de la plage d'intérêt. Cette nouvelle méthode permet de définir des régions d'intérêt potentiellement complexes dans l'espace d'entrée avec peu de paramètres et, en dehors, d'apprendre de manière non-paramétrique une transformation permettant d'améliorer la prédictivité du modèle sur la plage d'intérêt. Des simulations numériques montrent l'intérêt de la méthode pour l'optimisation bayésienne, où l'on est intéressé par les valeurs basses dans le cadre de la minimisation. De plus, la convergence théorique de la méthode est établie, sous certaines hypothèses.

**Title:** Improved Gaussian process modeling. Application to Bayesian optimization.
**Keywords:** Gaussian processes, Bayesian methods, model selection, kernel methods, optimization

**Abstract:** This manuscript focuses on Bayesian modeling of unknown functions with Gaussian processes. This task arises notably for industrial design, with numerical simulators whose computation time can reach several hours. Our work focuses on the problem of model selection and validation and goes in two directions.

The first part studies empirically the current practices for stationary Gaussian process modeling. Several issues on Gaussian process parameter selection are tackled. A study of parameter selection criteria is the core of this part. It concludes that the choice of a family of models is more important than that of the selection criterion. More specifically, the study shows that the regularity parameter of the Matérn covariance function is more important than the choice of a likelihood or cross-validation criterion. Moreover, the analysis of the numerical results shows that this parameter can be selected satisfactorily by the criteria, which leads to a practical recommendation. Then, particular attention is given to the numerical optimization of the likelihood criterion. Observing important inconsistencies between the different libraries available for Gaussian process modeling like Erickson et al. (2018), we propose elementary numerical recipes making it possible to obtain significant gains both in terms of likelihood and model accuracy. Finally, the analytical formulas for computing cross-validation criteria are revisited under a new angle and enriched with similar formulas for the gradients. This last contribution aligns the computational cost of a class of cross-validation criteria with that of the likelihood.

The second part presents a *goal-oriented* methodology. It is designed to improve the accuracy of the model in an (output) range of interest. This approach consists in relaxing the interpolation constraints on a relaxation range disjoint from the range of interest. We also propose an approach for automatically selecting the relaxation range. This new method can implicitly manage potentially complex regions of interest in the input space with few parameters. Outside, it learns non-parametrically a transformation improving the predictions on the range of interest. Numerical simulations show the benefits of the approach for Bayesian optimization, where one is interested in low values in the minimization framework. Moreover, the theoretical convergence of the method is established under some assumptions.

*À Erwan*

# Remerciements

Je souhaiterais en premier lieu remercier Emmanuel Vazquez et Julien Bect pour leur encadrement et leur soutien. Résumer tout ce qu'ils m'ont apporté serait sans doute trop long et remonterait à mon (très) lointain passé d'étudiant. Je suis conscient de la chance que j'ai eu d'avoir des encadrants si compétents et disponibles, qui ont passé un nombre incalculable d'heures à répondre à toutes mes questions sur le service de messagerie instantanée du laboratoire. J'ai pu aussi avoir le plaisir d'apprendre à vous connaître à travers toutes ces discussions, notamment autour de la musique.

Je voulais également remercier Paul Feliot pour son encadrement, son aide et pour m'avoir permis de mener cette thèse dans le sillage de la sienne. Un grand merci également à Jean Demange qui, bien qu'ayant rejoint l'équipe d'encadrement tardivement, a su comprendre les enjeux si rapidement pour faire aboutir ce travail et à Sébastien Da Veiga pour ses précieux conseils.

Mes remerciements vont également à Amandine Marrel, Rodolphe Le Riche, David Ginsbourger et Luc Pronzato pour avoir accepté de faire partie de mon jury de thèse et pour leurs lectures si détaillées et pertinentes de ce manuscrit, qui ont contribué à l'améliorer.

Je souhaite exprimer ma gratitude aux collègues de Safran Aircraft Engines m'ayant aidé ou fait profiter de leur bonne humeur : Samuel, Maxime, Josselyn, Benoit, Damien, Hanane, Marie, Matthieu, Stéphane, Pierre-Antoine...

Je tiens également à remercier les membres du L2S qui ont rendu ces années si sympathiques. Je pense notamment à Gilles, José, Elisabeth, Arthur, Charles, Laurent et Hani qui ont été de très bonne compagnie durant ces années. Il est impossible pour moi d'oublier les doctorants du laboratoire, qui ont tous participé à rendre mon cadre de travail si agréable. Ce fut un plaisir de discuter avec Romain et Subhasish, que je remercie également pour cette collaboration fructueuse que constitue le Chapitre 2 de cette thèse. J'ai également beaucoup profité des discussions passionnantes et animées avec Lucas et Fabien qui m'ont fait l'étalage de tous leurs talents, notamment d'imitateurs. Les derniers mots de ce paragraphes vont aux doctorants qui ont fait le chemin avec moi depuis le début : Abdelhak, Pierre, Arnaud, Bruno et enfin Manon qui a toujours été une oreille très attentive lors des phases de doute.

Enfin, ma plus grande reconnaissance va à Pauline, pour son indéfectible soutien et sans qui rien de tout ça n'aurait été possible, et à Erwan, qui illumine notre vie.

# Contents

# Part I

# Introduction

Figure 1: A fan module (left), composed of fan blades (middle) divided into airfoils (right).

# 1 . Context

## 1.1 . The exploration of numerical simulators

Engineers in the industry face the challenge of reaching design specifications using numerical models that are expensive-to-run computationally and even sometimes financially speaking. A technique for addressing this challenge is to substitute the expensive numerical model with a cheaper one, which serves as an approximation.

Such a surrogate model is built, or learned, from a small number of simulator runs. This approach is studied in the field of *computer experiments* (see, e.g., Currin et al., 1991, Sacks et al., 1989, Santner et al., 2003).

In many situations, it is not possible to obtain a model with good accuracy over the entire domain of its input parameters, and in this case we prefer to concentrate the simulations in regions of interest. To do so, it is usual to resort to sequential design strategies (see, e.g., Bect et al., 2012, 2019, Bernardo et al., 1992, Chevalier et al., 2014, Feliot et al., 2017, Jones et al., 1998, Villemonteix et al., 2009, and references therein), where one selects the next run of the simulator at each step by using the surrogate model.

## 1.2 . Industrial design: the example of a turbomachine fan blade

Let us now consider as an example the design of a turbomachine fan blade, which is one of the core businesses of Safran Aircraft Engines. A fan module, as shown in Figure 1, is a set of individual blades, which forms one of the most dimensioning components of an aircraft engine. Its role is to shape the air toward the primary flow and deliver the secondary flow acceleration. It is critical both for thrust and kerosene efficiency.

To design a fan blade, engineers start from a reference blade that is then parametrically deformed by considering the variations along the blade height of some geometrical quantities defining *airfoils*, shown in Figure 1. More precisely,

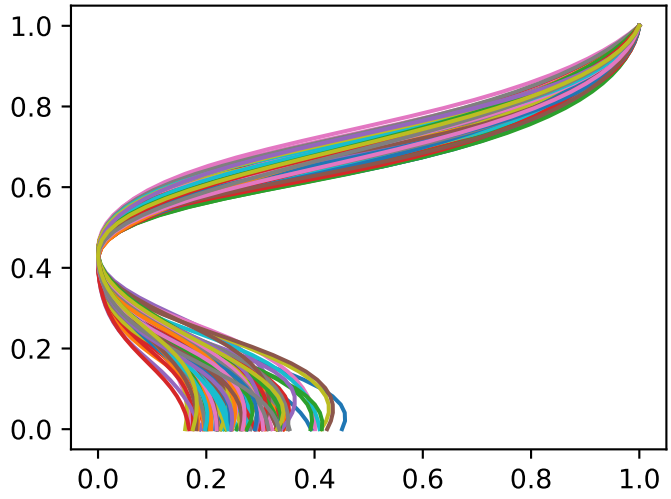



Figure 2: An example of design of experiments. The figure shows the variations of a geometrical quantity $\eta$ on the $x-$axis versus the height $h$ on the $y-$axis. The two axes are normalized.

the engineers fit the variations with some parametric curves, which are then used to deform the blade parametrically.

The parametrization must be flexible enough to propose new interesting geometries. However, maintaining a reasonable number of parameters is critical to limit the number of design variables and ensure deformations that are not too rough. An example of design of experiments produced during this thesis is shown in Figure 2.

A complex pipeline of numerical codes is then used to compute physical quantities of interest from the parametric deformations. The pipeline is multi-disciplinary: 1) the deformations are applied to the reference blade; 2) a first set of PDE solvers retrieve the deformations, the mechanical constraints, and the vibration frequencies of the blade; and 3) a Navier-Stokes solver infers the aerodynamical behavior of the ensemble during several flight phases. Using this pipeline, the objective of the engineers is then to find geometries that reach some performance specifications, which can be expressed, for example, as follows:

- Maximize the aircraft efficiency at a given flight phase
- Subject to:
  - Air flow constraints at some other flight phases
  - Mechanical constraints (frequency margin, fatigue resistance, ...).

The aforementioned specifications form a constrained (mono-objective) optimization problem. The physic of the system may be difficult to interpret, and the constraints are numerous. Manual exploration of the simulator is often time-consuming and inefficient. Consequently, automated sequential design of experiments techniques—presented in the next section—are very fruitful.

## 2 . Academic context and directions of research

### 2.1 . Probabilistic models and sequential design of experiments

A numerical simulator such as the one presented in Section 1 will be formalized as a continuous[1] function $f\colon \mathbb{X} \to \mathbb{R}^q$, where $\mathbb{X}$ is the simulator input space and $q$ is the number of simulator outputs. This manuscript will focus on parametric formulations, so we suppose that $\mathbb{X} \subset \mathbb{R}^d$. Moreover, the outputs of the simulator will be modeled independently, so we can also set $q = 1$ from now on.

When the evaluation of $f$ is costly, we want to construct an approximation of $f$ using a model. One usually requires models that can produce uncertainty estimates, since it is unlikely that the practitioner gathers enough data to neglect it. For this purpose, let $\mathscr{D} = \bigcup_{n\geq 1} \mathbb{X}^n \times \mathbb{R}^n$ be the set of possible finite observations of the values of $f$ on $\mathbb{X}$. We consider probabilistic models $\mathscr{M}\colon \mathscr{D} \to \mathscr{P}$, where $\mathscr{P}$ is a set of distributions over $\mathbb{R}^{\mathbb{X}}$ (endowed with the cylindrical $\sigma-$algebra). In this framework, $\mathscr{M}(D_n)$ is a predictive distribution for $f$ given $D_n \in \mathscr{D}$.

Given a model $\mathscr{M}$, stepwise uncertainty reduction (SUR) techniques (see, e.g., Bect et al., 2019, and references therein) form an important subclass of the sequential design strategies described in Section 1. In a nutshell, a SUR strategy consists in choosing the evaluations by optimizing a sampling criterion tailored to the task at hand. A broad variety of criteria is available in the literature (see, e.g., Bect et al., 2012, 2019, Chevalier et al., 2014, Feliot et al., 2017, Ginsbourger and Le Riche, 2010, Jones et al., 1998, Picheny et al., 2010, 2013, Villemonteix et al., 2009, and references therein).

The formalism of optimization adopted for the industrial case presented in Section 1.2 has received particular attention in the literature under the name of Bayesian optimization. One of the most ubiquitous sampling criteria for optimization is probably the expected improvement (EI) criterion (see, e.g., Mockus, 1975, Mockus et al., 1978), for which the overall optimization algorithm has been popularized under the name of Efficient Global Optimization (EGO) (Jones et al., 1998). For an $x \in \mathbb{X}$ and $D_n \in \mathscr{D}$, the EI criterion can be written as

$$\rho_n(x) = \mathsf{E}_{\xi \sim \mathscr{M}(D_n)} \left( \left( \xi(x_1) \wedge \cdots \wedge \xi(x_n) - \xi(x) \right)_+ \right), \tag{1}$$

in the case of a minimization problem. An illustration of the EI criterion is given in the next section.

[1] We suppose that continuity is inherited from the physical system.

Various extensions of the EI criterion have been proposed to account for batch evaluations, constraints and multiple objectives (see, e.g., Feliot et al., 2017, Ginsbourger et al., 2010, and references therein).

### 2.2 . Gaussian processes interpolation for computer experiments

The Bayesian framework is particularly appropriate for probabilistic modeling and consists in using a stochastic process $\xi : \Omega \times \mathbb{X} \to \mathbb{R}$ prior for $f$, where $\Omega$ is a probability space.

Gaussian processes (GP) are probably the most ubiquitous priors in the domain of computer experiments (see, e.g., Currin et al., 1988, Kitanidis, 1983), mainly because they make the inference tractable and form a flexible class of models (see, e.g., Rasmussen and Williams, 2006, Chapter 4). A GP $\xi$ is defined by an arbitrary mean function $\mu : \mathbb{X} \to \mathbb{R}$ and a symmetric positive (semi)definite covariance function $k : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$, but we shall use strictly positive definite covariance functions in this manuscript. Given the functions $\mu$ and $k$ and the data $D_n \in \mathscr{D}$, a GP yields a predictive distribution

$$\xi | D_n \sim \mathrm{GP}(\mu_n(x), k_n(x,x)), \tag{2}$$

with

$$\left\{ \begin{array}{rcl} \mu_n(x) & = & \mu(x) + k(x, \underline{x}_n) K_n^{-1} (\underline{Z}_n - \mu(\underline{x}_n)), \\ k_n(x,y) & = & k(x,y) - k(x, \underline{x}_n) K_n^{-1} k(y, \underline{x}_n)^{\mathsf{T}} \end{array} \right. \tag{3}$$

and where $\underline{x}_n = (x_1, \ldots, x_n)$, $k(x, \underline{x}_n) = (k(x, x_1), \ldots, k(x, x_n))$, $K_n$ is the matrix with entries $k(x_i, x_j)$, and $\underline{Z}_n = (\xi(x_1), \ldots, \xi(x_n))^{\mathsf{T}}$. A GP fit of a toy example is shown in Figure 3.

As mentioned above, the possibility of choosing $\mu$ and $k$ makes the framework of GPs flexible. For instance, a GP model can be significantly improved by using a mean function close to the target function. Moreover, Stein (1999) gives numerous arguments showing that different classes of positive definite covariance functions encode dramatically different assumptions.

Consequently, the predictive distribution (2) relies critically on $\mu$ and $k$, which must be chosen from the data and sometimes using expert knowledge. The standard approach is to select them from data within parametric families with maximum likelihood (ML) or cross-validation (CV) techniques (see, e.g., Currin et al., 1988). However, in our view, the literature is sparse about comparisons of such practices.

Furthermore, the parametric families used in practice are often stationary, i.e., with $\mu \equiv c \in \mathbb{R}$ and $k(x,y) = k_0(x-y)$ for some positive-definite function $k_0 : \mathbb{R}^d \to \mathbb{R}$. The stationary hypothesis is convenient for limiting the dimension of the parameter space. The most popular stationary family is probably the one of Matérn covariance functions (Matérn, 1986, Stein, 1999)

$$k_0(x) = \frac{2^{1-\nu}\sigma^2}{\Gamma(\nu)} \left(\sqrt{2\nu} h\right)^{\nu} \mathscr{K}_{\nu}\left(\sqrt{2\nu} h\right), \quad h = \left( \sum_{j=1}^{d} \frac{x_j^2}{\rho_j^2} \right)^{1/2}, \tag{4}$$

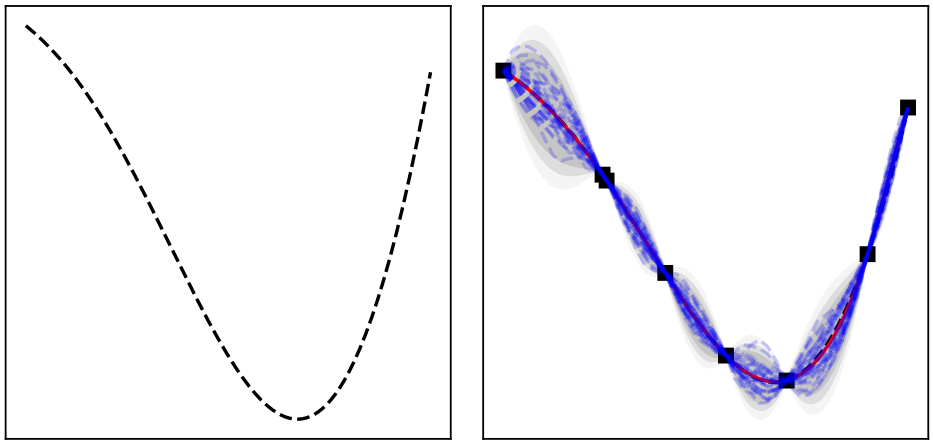


Figure 3: Left: a toy function to be minimized. Right: a Gaussian process fit. The red line represents the posterior mean, the blue lines the posterior sample paths, and the gray bands credible sets for several levels.

where $\Gamma$ is the Gamma function and $\mathscr{K}_\nu$ is the modified Bessel function of the second kind. Here $\sigma^2$ is the process variance, $\rho_i$ is a range parameter, and $\nu$ is a regularity parameter representing roughly the number of derivatives of the process. The parameters $\rho_i$s can be used to quantify the influence of the input variables (see, e.g., Marrel et al., 2009).

Unfortunately, $\mathscr{K}_\nu$ has a closed-form expression only when $\nu$ is infinite or can be written as $\chi + 1/2$ with $\chi \in \mathbb{N}$. Consequently, the parameter $\nu$ is usually enforced to one of the standard values $1/2$, $3/2$, $5/2$, or $\infty$ in practice. As pointed out by Stein (1999), these values refer to drastically different models. Figure 4 illustrates the effect of $\nu$ when fitting the toy example from Figure 3. Two kernels are used: a rough $\nu = 1/2$ Matérn covariance function and a much smoother $\nu = 17/2$ Matérn covariance function. The remaining parameters are selected by maximum likelihood estimation in both cases. Observe that the posterior is much more concentrated around the truth when using the smoother kernel. Consequently, the EI criterion clearly indicates the location of the global minimum, whereas it is much more spread if one uses the rough kernel.

Figure 5 introduces a slightly modified version of the previous toy function. Figure 6 shows a fit with the same two previous kernels. Observe that the rough kernel gives approximately the same results as on the original function. However, the fit with the smooth kernel is very different: its credible sets are now similar to those of the rough kernel, and its oscillating predictor induces a very unreasonable EI criterion.

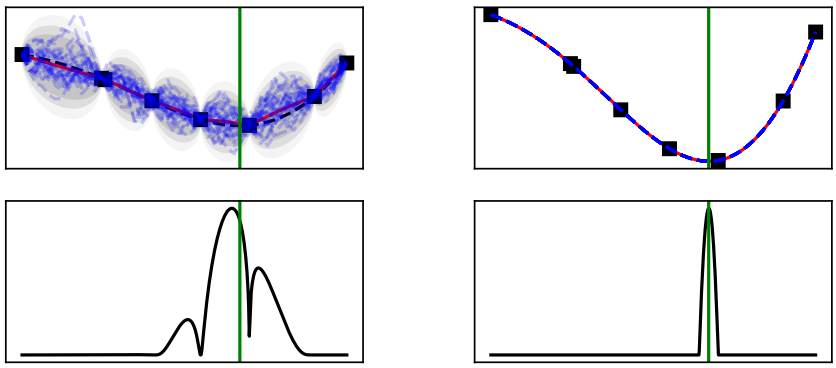


Figure 4: The EI criterion with two different Gaussian process priors. Top: Gaussian process fits of the toy function from Figure 3. Bottom: the EI criterion for minimization. Left: a fit with a rough kernel ($\nu = 1/2$); right: with a smooth kernel ($\nu = 17/2$).

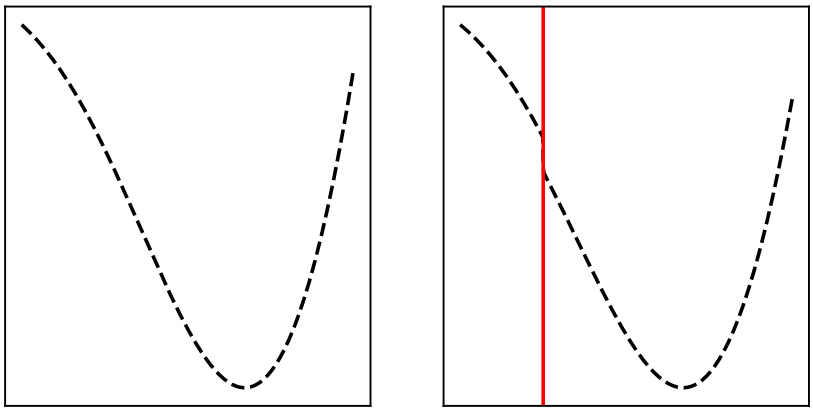


Figure 5: Left: the toy function from Figure 3. Right: the same function but with a perturbation taking the form of a jump around the $x$ value represented by the red horizontal line.

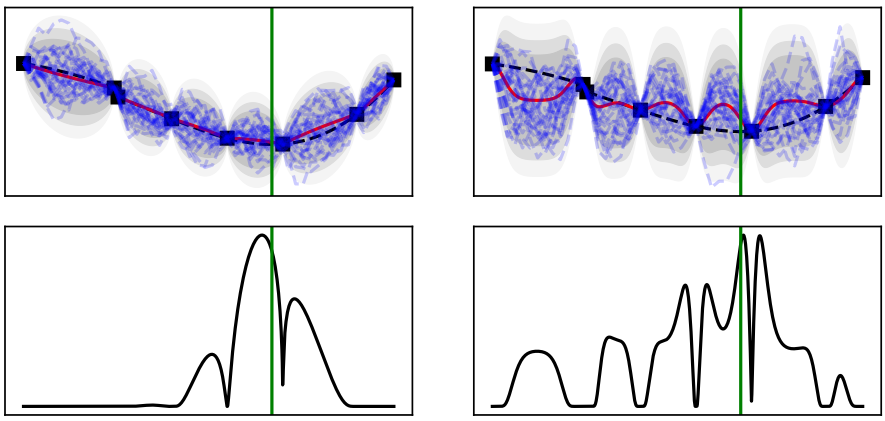


Figure 6: Same as Figure 4 but considering the perturbed toy minimization problem from Figure 5. The remaining parameters are still selected by maximum likelihood estimation.

Figure 4 and Figure 6 show that choosing among stationary models can affect significantly the accuracy. In Figure 6, a rough kernel works best in the big picture but is not very satisfying. Indeed, the effect of the perturbation is global even if it lies in one location. Far from the perturbation, a desirable fit would be as accurate as with the smooth model on the original function. Techniques for localizing Gaussian process predictions have been largely considered by researchers (see, e.g., Gramacy and Lee, 2008). However, the region where the function misbehaves can potentially be very complicated, especially when the dimension is high.

A research track in this manuscript is motivated by the following observation. Observe in Figure 6 that the perturbation is localized in the higher range of the function. If the goal is the minimization of $f$, then common sense suggests that it is important to be precise on the lower range of the function. Conversely, modeling precisely the higher range is probably not so critical.

### 2.3 . Problem statement

As we saw in Section 2.2, stationary GP modeling is a widespread practice, but selecting the parameters of a stationary model is critical. In addition, the stationary hypothesis is sometimes untenable. However, finding a trade-off between computational cost and expressiveness of a non-stationary model is challenging. The path of goal-oriented Gaussian process modeling seems promising and has seldom been explored. Therefore, this manuscript intends to make contributions on two topics:

- Improving the current practices for stationary GP modeling for prediction and optimization
- Going beyond stationary models for Bayesian optimization while providing

turnkey solutions for practitioners

Regarding the second topic, we want to take the path of goal-oriented modeling.

## 3 . Outline of the manuscript and contributions

The manuscript is organized into three parts.

Part II presents empirical contributions to parameter selection for stationary GP models. More precisely, Chapter 1 presents a benchmark for the choice of the model and the selection criterion for prediction and optimization. Some selection criteria are reviewed, then numerical experiments suggest that: 1) maximum likelihood provides performances that are better than or comparable to those of other (cross-validation) criteria; 2) the choice of the regularity parameter of a Matérn covariance function is a more critical factor; and 3) that the regularity parameter can be successfully chosen using the criteria. Furthermore, a comparison with models using the test set shows that there is only little room for improvement with respect to maximum likelihood estimation. Then, Chapter 2 presents numerical investigations about optimizing the likelihood for Gaussian processes interpolation. We show that the numerical noise amplified by the condition number of the covariance matrix disturbs standard gradient-based optimizers. We propose some simple numerical recipes to mitigate this issue. Numerical experiments are conducted and show significant benefits both regarding the likelihood values and the prediction errors. Finally, Chapter 3 revisits well-known fast cross-validation formulas for Gaussian process regression and attempts to make some clarifications from a Bayesian point of view. The fast formulas are then extended to compute the gradients, aligning the computational cost of a class of cross-validation criteria to the one of the likelihood.

Part III presents our contribution to goal-oriented modeling for Bayesian optimization. After reviewing the existing methods for non-stationary modeling, we propose a general goal-oriented framework for Gaussian process interpolation, designed to improve the accuracy in a range of interest. The methodology improves the precision in the range of interest by learning a transformation non-parametrically in a relaxation range jointly with the GP parameters. Given the relaxation range, the method maintains a reasonable computational cost, making it possible to use adaptive strategies—which can be seen as a contribution of independent interest—for selecting the relaxation range, i.e., for detecting which values are helpful or not to predict the range of interest. Numerical experiments demonstrate the interest of the approach, illustrating the relevance of the estimated transformations and showing more generally its benefits for Bayesian optimization if one targets low values. In particular, the technique is applied in Figure 7 to the perturbed problem shown in Figure 5: it makes it possible to recover the precision of the smooth kernel on the original problem. We also provide a theoretical convergence result of the resulting optimization algorithm and error bounds that suggest

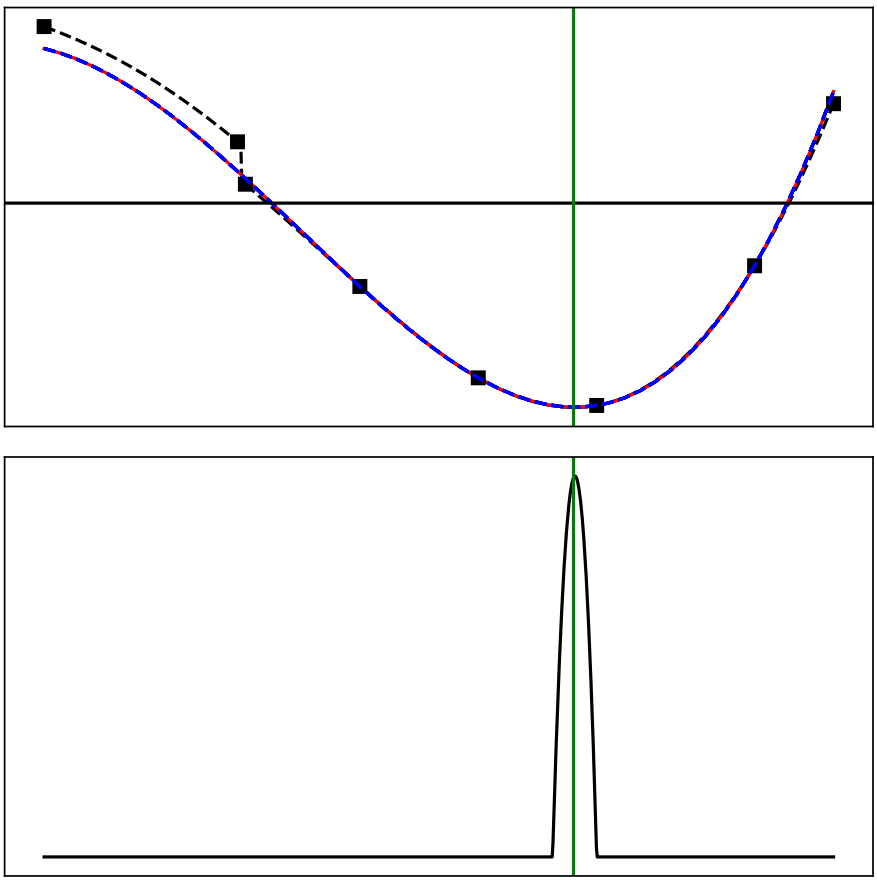


Figure 7: Same as Figure 6 but using the proposed goal-oriented methodology with the smooth kernel. The black line represents a threshold that has been selected automatically.

an improved fit in the range of interest when the function lies in the reproducing kernel Hilbert space attached to the (fixed) covariance function.

Finally, Part IV summarizes the contributions, the limitations, and the perspectives for future works.

## 4 . Communications

Chapter 1 is mostly a reproduction of Petit et al. (2021a). Chapter 2 is a reproduction of Basak et al. (2021). Chapter 3 is an article in preparation with Julien Bect and Emmanuel Vazquez extending Petit et al. (2020a). Finally, Chapter 4 is mostly a reproduction of Petit et al. (2022b).

**Preprints**

S. J. Petit, J. Bect, P. Feliot, and E. Vazquez. Model parameters in Gaussian

process interpolation: an empirical study of selection criteria. *Submitted to the SIAM/ASA Journal on Uncertainty Quantification*, 2021a. URL https://arxiv.org/abs/2107.06006

S. J. Petit, J. Bect, and E. Vazquez. Relaxed Gaussian process interpolation: a goal-oriented approach to Bayesian optimization. 2022b. URL https://arxiv.org/abs/2107.06006

**Communications with proceedings**

S. J. Petit, J. Bect, S. Da Veiga, P. Feliot, and E. Vazquez. Towards new cross-validation-based estimators for Gaussian process regression: efficient adjoint computation of gradients. In *52èmes Journées de Statistique de la SFdS (JdS 2020)*, 2020a

S. Basak, S. J. Petit, J. Bect, and E. Vazquez. Numerical issues in maximum likelihood parameter estimation for Gaussian process regression. In *7th International Conference on machine Learning, Optimization and Data science*, 2021

**Communications without proceedings**

S. J. Petit, J. Bect, S. Da Veiga, P. Feliot, and E. Vazquez. Presentation: optimisation bayésienne avec application à la conception d'un aubage fan. In *Séminaire EDF/CEA doctorants en statistiques et incertitudes*, 2019

S. J. Petit, J. Bect, P. Feliot, and E. Vazquez. Poster: Gaussian process model selection for computer experiments. In *Journées annuelles du GdR MASCOT NUM (MASCOT NUM 2020)*, 2020b

S. J. Petit, J. Bect, and E. Vazquez. Presentation: Relaxed Gaussian process interpolation: a goal-oriented approach to Bayesian optimization. In *Journées annuelles du GdR MASCOT NUM (MASCOT NUM 2022)*, 2022a

# Part II

# Choosing a Gaussian process prior

# 1 - Model parameters in Gaussian process interpolation: an empirical study of selection criteria

This chapter is a reproduction of Petit et al. (2021a) with few modifications and enriched with the additional numerical experiments from Section 1.6. A Supplementary Material (Petit et al., 2021b) with a complete description of the numerical results is available.

S. J. Petit, J. Bect, P. Feliot, and E. Vazquez. Model parameters in Gaussian process interpolation: an empirical study of selection criteria. *Submitted to the SIAM/ASA Journal on Uncertainty Quantification*, 2021a. URL https://arxiv.org/abs/2107.06006

S. J. Petit, J. Bect, P. Feliot, and E. Vazquez. Model parameters in Gaussian process interpolation: an empirical study of selection criteria: Supplementary Material. 2021b. URL https://hal-centralesupelec.archives-ouvertes.fr/hal-03285513v2/file/spetit-gpparam-suppmat.pdf

## 1.1 . Introduction

Regression and interpolation with Gaussian processes, or kriging, is a popular statistical tool for non-parametric function estimation, originating from geostatistics and time series analysis, and later adopted in many other areas such as machine learning and the design and analysis of computer experiments (see, e.g., Rasmussen and Williams (2006), Santner et al. (2003), Stein (1999) and references therein). It is widely used for constructing fast approximations of time-consuming computer models, with applications to calibration and validation Bayarri et al. (2007), Kennedy and O'Hagan (2001), engineering design Forrester et al. (2008), Jones et al. (1998), Bayesian inference Calderhead et al. (2009), Wilkinson (2014), and the optimization of machine learning algorithms Bergstra et al. (2011)—to name but a few.

A Gaussian process (GP) prior is characterized by its mean and covariance functions. They are usually chosen within parametric families (for instance, constant or linear mean functions, and Matérn covariance functions), which transfers the problem of choosing the mean and covariance functions to that of selecting parameters. The selection is most often carried out by optimization of a criterion that measures the goodness of fit of the predictive distributions, and a variety of such criteria—the likelihood function, the leave-one-out (LOO) squared-prediction-error criterion (hereafter denoted by LOO-SPE), and others—is available from the literature. The search for arguments to guide practitioners in the choice of one

particular criterion is the main motivation of this study.

As a necessary parenthesis, note that the fully Bayesian statistician does not select a particular value for the parameters, but chooses instead to average the predictions over a *posterior* distribution on the parameters Handcock and Stein (1993). This approach may be preferred for more robust predictions (see, e.g., Benassi et al. (2011)), but comes at a higher computational cost. For this reason, the present chapter will set aside the fully Bayesian approach and concentrate on the plugin approach, where only one parameter value is chosen to carry out predictions.

The two most popular methods for parameter selection are maximum likelihood (ML) and cross-validation (CV) based on LOO criteria, which were introduced in the field of computer experiments by the seminal work of Currin et al. (1988). Since then, despite a fairly large number of publications dealing with ML and CV techniques for the selection of a GP model, the literature has remained in our view quite sparse about the relative merits of these methods, from both theoretical and empirical perspectives.

For instance, in the framework of interpolation and infill asymptotics, where observations accumulate in some bounded domain, Ying (1991) and Zhang (2004) show that some combinations of the parameters of a Matérn covariance function can be estimated consistently by ML. Again in the case of infill asymptotics, with the additional assumption of a one-dimensional GP with an exponential covariance function, Bachoc et al. (2016) show that estimation by LOO is also consistent. (Similar results exist in the expanding asymptotic framework, where observations extend over an ever-increasing horizon.)

The practical consequences of the aforementioned results are somewhat limited in our view because practitioners are primarily interested in the quality of the predictions. Knowing that the parameters of a GP can be estimated consistently is intellectually reassuring, but may be considered of secondary importance. These results are indeed about the statistical model itself but they say little about the prediction properties of the model. Besides, there does not exist at present, to our knowledge, some theoretically-based evaluation of the relative performances of ML and CV selection criteria under infill asymptotics.

Turning now to empirical comparisons of selection criteria for GP interpolation, the first attempt in the domain of computer experiments can be traced back to the work of Currin et al. (1988, 1991). The authors introduce CV and ML—which can be seen as a special kind of cross-validation—and present some simple experiments using tensorized covariance functions, from which they conclude that, "Of the various kinds of cross-validation [they] have tried, maximum likelihood seems the most reliable". Additional experiments have been conducted by several authors, but no consensus emerges from these studies: Bachoc (2013b), Martin and Simpson (2003), Sundararajan and Keerthi (2001) conclude in favor of CV whereas Martin and Simpson (2005) advocate ML.

These studies are limited to a rather small number of test functions and covariance functions, which may explain the discrepancy in the conclusions of those experiments. In particular, only Bachoc (2013b) considers the popular and versatile Matérn covariance functions. Moreover, most studies focus only on the accuracy of the posterior mean—only Sundararajan and Keerthi (2001) and Bachoc (2013b) provide results accounting for the quality of the posterior variance—whereas the full posterior predictive distribution is used in most GP-based methods (see, e.g., Chevalier et al. (2014), Jones et al. (1998)).

This chapter presents two main contributions. First, we improve upon the results of the literature by providing an empirical ranking of selection criteria for GP interpolation, according to several metrics measuring the quality of posterior predictive distributions on a large set of test functions from the domain of computer experiments. To this end, we base our study on the general concept of scoring rules Gneiting and Raftery (2007), Zhang and Wang (2010), which provides an effective framework for building selection and validation criteria. We also introduce a notion of extended likelihood criteria, borrowing an idea from Fasshauer and co-authors Fasshauer (2011), Fasshauer et al. (2009) in the literature of radial basis functions.

Second, we provide empirical evidence that the choice of an appropriate family of models is often more important—and sometimes much more important, especially when the size of the design increases—than the choice of a particular selection criterion (e.g., likelihood versus LOO-SPE). More specifically, in the case of the Matérn family, this leads us to assess, and ultimately recommend, the automatic selection of a suitable value of the regularity parameter, against the common practice of choosing beforehand an arbitrary value of this parameter.

The chapter is organized as follows. Section 1.2 briefly recalls the general framework of GP regression and interpolation. Section 1.3 reviews selection criteria for GP model parameters. After recalling the general notion of scoring rules, we present a broad variety of selection criteria from the literature. Section 1.4 presents experimental results on the relative performances of these criteria. Leveraging the findings from Section 1.4, Section 1.6 presents a benchmark on the choice of a family of models for Bayesian optimization. Section 1.5 presents our conclusions and perspectives.

## 1.2 . General framework

Let us consider the general GP approach for a scalar-valued deterministic computer code with input space $\mathbb{X} \subseteq \mathbb{R}^d$. The output of the computer code $z : \mathbb{X} \to \mathbb{R}$ is modeled by a random function $(Z(x))_{x \in \mathbb{X}}$, which, from a Bayesian perspective, represents prior knowledge about $z$. If we assume that $Z(\cdot)$ is observed on a design $\mathbb{X}_n = \{x_1, \ldots, x_n\}$ of size $n$, the data corresponds to a sample of the random vector $\mathbf{Z}_n = (Z(x_1), \ldots, Z(x_n))^\mathsf{T}$.

The GP assumption makes it possible to easily derive posterior distributions.

Table 1.1: Popular Matérn subfamilies

| | $\nu = \frac{1}{2}$ | $\nu = \frac{3}{2}$ | $\nu = \frac{5}{2}$ | $\nu = +\infty$ |
|---|---|---|---|---|
| a.k.a. | exponential | | | squared exponential |
| $k_\theta(x,y)$ | $\sigma^2 e^{-h}$ | $\sigma^2(1+\sqrt{3}h)e^{-\sqrt{3}h}$ | $\sigma^2(1+\sqrt{5}h+\frac{5h^2}{3})e^{-\sqrt{5}h}$ | $\sigma^2 e^{-\frac{h^2}{2}}$ |

More precisely, it is assumed that $Z(\cdot)$ is a Gaussian process, with (prior) mean $\mathrm{E}(Z(x)) = \sum_{l=1}^{L} \beta_l \phi_l(x)$, where the $\beta_1, \ldots, \beta_L$ are unknown regression parameters and $\phi_1, \ldots, \phi_l$ are known regression functions, and with (prior) covariance

$$\mathrm{cov}\,(Z(x), Z(y)) = k_\theta(x,y),$$

where $\theta \in \Theta \subseteq \mathbb{R}^q$ is a vector of *parameters*. Throughout the chapter, the covariance matrix of $\mathbf{Z}_n$ will be denoted by $\mathbf{K}_\theta$. We assume for simplicity that the prior mean of $Z(\cdot)$ is zero (hence, $L = 0$), which is a common practice when data are centered.

One of the most popular covariance functions for GP regression is the anisotropic stationary Matérn covariance function Matérn (1986) popularized by Stein (1999):

$$k_\theta(x,y) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}h\right)^\nu \mathscr{K}_\nu\left(\sqrt{2\nu}h\right), \quad h = \left(\sum_{j=1}^{d} \frac{(x_j - y_j)^2}{\rho_j^2}\right)^{1/2}, \qquad (1.1)$$

where $\Gamma$ is the Gamma function, $\mathscr{K}_\nu$ is the modified Bessel function of the second kind, and $\theta$ denotes the vector of parameters $\theta = (\sigma^2, \rho_1, \ldots, \rho_d, \nu) \in \Theta = ]0,\infty[^{d+2}$. The parameter $\sigma^2$ is the variance of $Z(\cdot)$, the $\rho_i$s are range parameters which characterize the typical correlation length on each dimension, and $\nu$ is a regularity parameter, whose value controls the mean-square differentiability of $Z(\cdot)$. Recall (see Table 1.1) that the Matérn covariance function with $\nu = 1/2$ corresponds to the so-called exponential covariance function, and the limiting case $\nu \to \infty$ can be seen as the "squared exponential" (also called Gaussian) covariance function.

Because $\mathscr{K}_\nu$ has a closed-form expression when $\nu - \frac{1}{2}$ is an integer, and is more expensive to evaluate numerically in other cases, most implementations choose to restrict $\nu$ to half-integer values. Moreover, a widespread practice (in applications and research papers) consists in selecting a particular value for $\nu$ (e.g., $\nu = 1/2$, $\nu = 3/2\ldots$ or the limiting case $\nu \to \infty$), once and for all.

Since the family of Matérn covariance functions is widely used in practice, we focus exclusively on this family in this work. We believe that the conclusions of the present study would not be altered significantly if other families of covariance functions (e.g., the compactly supported covariance functions proposed by Wendland (1995)) were considered.

Once a GP model has been chosen, the framework of Gaussian process regression allows one to build a predictive distribution $\mathscr{N}(\mu_\theta(x), \sigma_\theta^2(x))$ for an unobserved

$Z(x)$ at $x \in \mathbb{R}^d$, where

$$\begin{cases} \mu_\theta(x) &= \mathbf{k}_\theta^*(x)^\mathsf{T} \mathbf{K}_\theta^{-1} \mathbf{Z}_n, \\ \sigma_\theta^2(x) &= k_\theta(x,x) - \mathbf{k}_\theta^*(x)^\mathsf{T} \mathbf{K}_\theta^{-1} \mathbf{k}_\theta^*(x) \end{cases} \tag{1.2}$$

with $\mathbf{k}_\theta^*(x) = (k_\theta(x,x_1), \ldots, k_\theta(x,x_n))^\mathsf{T}$. More generally, predictive distributions can be built for a larger range of quantities of interest such as joint observations, derivatives, integrals or excursions of $Z$ above a given threshold.

Using this framework, the user obtains a family of Bayesian procedures, indexed by $\theta$, to perform predictions about the unknown computer code at hand, and must choose a member of the family that will hopefully provide good predictive performances.

## 1.3 . Selection of a GP model from a parameterized family

### 1.3.1 . Scoring rules

Goodness-of-fit criteria for probabilistic predictions have been studied in the literature under the name of scoring rules by Gneiting and Raftery (2007). A (negatively oriented) scoring rule is a function $S(\cdot\,; z) : \mathscr{P} \to \mathbb{R} \cup \{-\infty, +\infty\}$, acting on a class $\mathscr{P}$ of probability distributions on $\mathbb{R}$, such that $S(P; z)$ assigns a loss for choosing a predictive distribution $P \in \mathscr{P}$, while observing $z \in \mathbb{R}$. Scoring rules make it possible to quantify the quality of probabilistic predictions.

**Example 1 (squared prediction error)** *Denoting by $\mu$ the mean of a predictive distribution $P$, the squared prediction error*

$$S^{\mathrm{SPE}}(P; z) = (z - \mu)^2 \tag{1.3}$$

*accounts for the deviation of $z$ from $\mu$. Note that $S^{\mathrm{SPE}}$ ignores subsequent moments and therefore predictive uncertainties.*

**Example 2 (negative log predictive density)** *Denoting by $p$ the probability density of $P$,*

$$S^{\mathrm{NLPD}}(P; z) = -\log(p(z)) \tag{1.4}$$

*tells how likely $z$ is according to $P$. Bernardo (1979) shows that any (proper) scoring rule that depends on $p(z)$ (and only on $p(z)$) can be reduced to $S^{\mathrm{NLPD}}$.*

**Example 3 (continuous ranked probability score)** *Let $U$ and $U'$ be two independent random variables with distribution $P$. The CRPS quantifies the deviation of $U$ from $z$:*

$$S^{\mathrm{CRPS}}(P; z) = \mathsf{E}(|U - z|) - \frac{1}{2}\mathsf{E}(|U - U'|)\,. \tag{1.5}$$

*Székely and Rizzo (2005) show that $S^{\mathrm{CRPS}}(P; z) = \int (P(U \leq u) - \mathbb{1}_{z \leq u})^2 \,\mathrm{d}u$, the CRPS can also be seen as a (squared) distance between the empirical cumulative distribution $u \mapsto \mathbb{1}_{z \leq u}$ and the cumulative distribution of $P$.*

Table 1.2: Scoring rules behavior as $|\mu - z| \ll 1$.

| | $\sigma \ll \lvert\mu - z\rvert$ | $\sigma \simeq \lvert\mu - z\rvert$ | $\sigma \gg \lvert\mu - z\rvert$ |
|---|---|---|---|
| $S^{\mathrm{SPE}}(P; z)$ | $0$ | $0$ | $0$ |
| $S^{\mathrm{NLPD}}(P; z)$ | $+\infty$ | $-\infty$ | $\log(2\pi\sigma)$ |
| $S^{\mathrm{CRPS}}(P; z)$ | $0$ | $0$ | $\propto \sigma$ |
| $S^{\mathrm{IS}}_{1-\alpha}(P; z)$ | $0$ | $0$ | $\propto \sigma$ |

*Note that if absolute values in* (1.5) *are replaced by squared values, then* $S^{\mathrm{SPE}}$ *is recovered. The CRPS can also be extended to the so-called energy and kernel scores Gneiting and Raftery (2007) by observing that* $(x, y) \mapsto |x - y|$ *is a conditionally negative kernel.*

**Example 4 (interval score)** *The interval scoring rule at level* $1 - \alpha$ *is defined, for* $\alpha \in ]0, 1[$*, by*

$$S^{\mathrm{IS}}_{1-\alpha}(P; z) = (u - l) + \frac{2}{\alpha}(l - z)\mathbb{1}_{z \leq l} + \frac{2}{\alpha}(z - u)\mathbb{1}_{z > u} \tag{1.6}$$

*where* $l$ *and* $u$ *are the* $\alpha/2$ *and* $1 - \alpha/2$ *quantiles of* $P$*. The first term penalizes large intervals, while the second and third terms penalize intervals not containing* $z$*.*

When the predictive distribution $P$ is Gaussian, which is the case when $P$ is the posterior distribution of a GP $Z$ at a given point, the aforementioned scoring rules all have closed-form expressions. More precisely, for $P = \mathcal{N}(\mu, \sigma^2)$, we simply have $S^{\mathrm{SPE}}(P; z) = (z - \mu)^2$ and $S^{\mathrm{NLPD}}(P; z) = \frac{1}{2}\log 2\pi\sigma^2 + \frac{1}{2}(z - \mu)^2/\sigma^2$. $S^{\mathrm{IS}}_{1-\alpha}$ can be obtained by taking the standard expressions of the $\alpha/2$ and $1 - \alpha/2$ quantiles of $P$, and it can be shown that

$$S^{\mathrm{CRPS}}(P; z) = \sigma\Big(\frac{z - \mu}{\sigma}\Big(2\Phi\big(\frac{z - \mu}{\sigma}\big) - 1\Big) + 2\phi\big(\frac{z - \mu}{\sigma}\big) - \frac{1}{\sqrt{\pi}}\Big),$$

where $\phi$ and $\Phi$ stand respectively for the probability density function and the cumulative distribution function of the standard Gaussian distribution.

Note that all aforementioned scoring rules penalize large values of $|z - \mu|$. When $|z - \mu| \ll 1$ different scoring rules yield different penalties, as reported in Table 1.2.

### 1.3.2 . Selection criteria

## Leave-one-out selection criteria

Scoring rules make it possible to build criteria for choosing the parameters of a GP. More precisely, to select $\theta$ based on a sample $Z_1, \ldots, Z_n$, one can minimize the mean score

$$J^S_n(\theta) = \frac{1}{n}\sum_{i=1}^{n} S(P_{\theta,-i}; Z_i), \tag{1.7}$$

where $S$ is a scoring rule and $P_{\theta,-i}$ denotes the distribution of $Z_i$ conditional on the $Z_j$s, for $1 \leq j \leq n,\ j \neq i$, indexed by $\theta$.

Selection criteria written as (1.7) correspond to the well-established leave-one-out (LOO) method, which has been proposed in the domain of computer experiments by Currin et al. (1988), and is now used in many publications (see, e.g., Rasmussen and Williams (2006), and also Zhang and Wang (2010), who formally adopt the point of view of the scoring rules, but for model validation instead of parameter selection).

**Efficient computation of predictive distributions** Leave-one-out predictive densities can be computed using fast algebraic formulas Craven and Wahba (1979), Dubrule (1983). More precisely, the predictive distribution $P_{\theta,-i}$ is a normal distribution $\mathcal{N}(\mu_{\theta,-i}, \sigma^2_{\theta,-i})$ with

$$\mu_{\theta,-i} = Z(x_i) - \frac{(\mathbf{K}_\theta^{-1}\mathbf{Z}_n)_i}{\mathbf{K}^{-1}_{\theta,i,i}} \quad \text{and} \quad \sigma^2_{\theta,-i} = \frac{1}{\mathbf{K}^{-1}_{\theta,i,i}}. \tag{1.8}$$

Furthermore, Petit et al. (2020a) show that, using reverse-mode differentiation, it is possible to compute mean scores $J_n$ and their gradients with a $O(n^3 + dn^2)$ computational cost, which is the same computational complexity as for computing the likelihood function and its gradient (see, e.g., Rasmussen and Williams (2006)).

**The particular case of LOO-SPE** The LOO selection criterion

$$J_n^{\mathrm{SPE}}(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\mu_{\theta,-i} - Z(x_i))^2\,, \tag{1.9}$$

based on the scoring rule (1.3) will be referred to as LOO-SPE. This criterion, also called prediction sum of squares (PRESS) Allen (1974), Wahba (1990) or LOO squared bias Currin et al. (1988), is well known in statistics and machine learning, and has been advocated by some authors Bachoc (2013b, 2018), Martin and Simpson (2003), Santner et al. (2003), Sundararajan and Keerthi (2001) to address the case of "misspecified" covariance functions.

However, note that $\sigma^2$ cannot be selected using $J_n^{\mathrm{SPE}}$. When $J_n^{\mathrm{SPE}}$ is used, $\sigma^2$ is generally chosen (see, e.g., Bachoc (2013b), Currin et al. (1988) and Remark 5) to satisfy

$$\frac{1}{n}\sum_{i=1}^{n}\frac{(Z(x_i) - \mu_{\theta,-i})^2}{\sigma^2_{\theta,-i}} = 1\,, \tag{1.10}$$

which will be referred to as Cressie's rule for $\sigma^2$, in reference to the claim by Cressie (1993) that (1.10) should hold approximately for a good GP model.

**Other scoring rules for LOO** The selection criteria using the NLPD scoring rule (1.4) and the CRPS scoring rule (1.5) will be referred to as the LOO-NLPD and LOO-CRPS criteria, respectively. The LOO-NLPD criterion has been called preditive deficiency in Currin et al. (1988), and Geisser's surrogate Predictive Probability (GPP) in Sundararajan and Keerthi (2001). The LOO-CRPS criterion has been considered in Zhang and Wang (2010) as a criterion for model validation (see also Demay et al. (2021) for an application to model selection), and more recently Petit et al. (2020a,b) as a possible criterion for parameter selection as well.

**Remark 5** *Note that Cressie's rule* (1.10) *can be derived by minimizing the LOO-NLPD criterion with respect to* $\sigma^2$*.*

**Remark 6** *In order to limit the number of selection criteria under study, the interval scoring rule is only used for validation in this work.*

## Maximum likelihood and generalizations

We can safely say that the most popular method for selecting $\theta$ from data is maximum likelihood estimation—and related techniques, such as restricted maximum likelihood estimation. The ML estimator is obtained by maximizing the likelihood function or, equivalently, by minimizing the negative log-likelihood (NLL) selection criterion. Denoting by $p_\theta(\mathbf{Z}_n)$ the joint density of $\mathbf{Z}_n$, the NLL selection criterion may be written as

$$J_n^{\mathrm{NLL}}(\theta) = -\log(p_\theta(\mathbf{Z}_n)) = \frac{1}{2}\left(n\log(2\pi) + \log\det\mathbf{K}_\theta + \mathbf{Z}_n^\mathsf{T}\mathbf{K}_\theta^{-1}\mathbf{Z}_n\right). \tag{1.11}$$

As pointed out by Currin et al. (1988), the NLL criterion is closely related to the LOO-NLPD criterion, through the identity

$$J_n^{\mathrm{NLL}}(\theta) = -\log(p_\theta(Z(x_1)) - \sum_{i=2}^{n}\log(p_\theta(Z(x_i) \mid Z(x_1),\ldots,Z(x_{i-1}))),$$

where the predictive distributions of the $Z(x_i)$s given the $Z(x_1),\ldots,Z(x_{i-1})$ explicitly appear.

One can minimize (1.11) in closed-form with respect to $\sigma^2$, given other parameters. Writing $\mathbf{K}_\theta = \sigma^2\mathbf{R}_\theta$ and canceling $\partial J_n^{\mathrm{NLL}}(\theta)/\partial\sigma^2 = \left(n\sigma^2 - \mathbf{Z}_n^\mathsf{T}\mathbf{R}_\theta^{-1}\mathbf{Z}_n\right)/(2\sigma^2)$ yields

$$\sigma^2_{\mathrm{NLL}} = \frac{1}{n}\mathbf{Z}_n^\mathsf{T}\mathbf{R}_\theta^{-1}\mathbf{Z}_n, \tag{1.12}$$

which will be referred to as the profiling rule for $\sigma^2$.

Injecting (1.12) into (1.11) yields a *profiled likelihood* selection criterion, that can be written as

$$J_n^{\mathrm{PL}}(\theta) = \log\sigma^2_{\mathrm{NLL}} + \frac{1}{n}\log\det\mathbf{R}_\theta = \log\left(\frac{1}{n}\mathbf{Z}_n^\mathsf{T}\mathbf{R}_\theta^{-1}\mathbf{Z}_n\right) + \frac{1}{n}\log\det\mathbf{R}_\theta. \tag{1.13}$$

Following Fasshauer and co-authors Fasshauer (2011), Fasshauer et al. (2009), we consider now a family of selection criteria that extends (1.11). Using the factorization $\mathbf{R}_\theta = \mathbf{Q}\Lambda\mathbf{Q}^\mathsf{T}$, where $\mathbf{Q} = (\mathbf{q}_1, \ldots, \mathbf{q}_n)$ is an orthogonal matrix of (orthonormal) eigenvectors and $\Lambda = \operatorname{diag}(\lambda_1, \cdots, \lambda_n)$, notice that

$$\exp\left(J_n^{\mathrm{PL}}(\theta)\right) = \frac{1}{n}\mathbf{Z}_n^\mathsf{T}\mathbf{R}_\theta^{-1}\mathbf{Z}_n \cdot (\det \mathbf{R}_\theta)^{1/n} \propto \left(\sum_{i=1}^n \left(\boldsymbol{q}_i^\mathsf{T}\mathbf{Z}_n\right)^2/\lambda_i\right)\left(\prod_{i=1}^n \lambda_j\right)^{1/n}. \quad (1.14)$$

This suggests a generalization of the likelihood criterion that we shall call Fasshauer's Hölderized likelihood (HL), defined as

$$J_n^{\mathrm{HL},\,p,q}(\theta) = \left(\sum_{i=1}^n (\boldsymbol{q}_i^\mathsf{T}\mathbf{Z}_n)^2/\lambda_i^p\right)^{1/p}\left(\frac{1}{n}\sum_{j=1}^n \lambda_j^q\right)^{1/q}, \quad (1.15)$$

with $q \in [-\infty, +\infty]$, and $p \in \mathbb{R}\setminus\{0\}$, and where $\sigma^2$ can be chosen using the rules (1.10) or (1.12), since $J_n^{\mathrm{HL},\,p,q}(\theta)$ does not depend on $\sigma^2$. Owing to the standard property of generalized means $\left(\frac{1}{n}\sum_{i=1}^n x_i^q\right)^{\frac{1}{q}} \xrightarrow{q\to 0} \sqrt[n]{x_1\cdots x_n}$), (1.14) is recovered by taking $p = 1$ and letting $q \to 0$. Moreover, two other known selection criteria can be obtained for particular values of $p$ and $q$, as detailed below.

**Generalized cross-validation** Taking $p = 2$ and $q = -1$ in (1.15) yields the generalized cross-validation (GCV) criterion

$$J_n^{\mathrm{GCV}}(\theta) = n^{-1}\left(J_n^{\mathrm{HL},2,-1}(\theta)\right)^2,$$

which was originally proposed as a rotation-invariant version of PRESS Golub et al. (1979) for linear models. It has also been shown to be efficient for the selection of the smoothing parameter of polyharmonic splines Wahba (1990) and for the selection of the degree of a spline Wahba and Wendelberger (1980).

The GCV selection criterion is a weighted SPE criterion, which can also be written as

$$J_n^{\mathrm{GCV}}(\theta) = \frac{1}{n}\sum_{i=1}^n w_i^2(\theta)(Z(x_i) - \mu_{\theta,-i})^2, \quad w_i(\theta) = \frac{\tilde{\sigma}^2}{\sigma_{\theta,-i}^2}, \quad (1.16)$$

with $\tilde{\sigma}^2 = \left(\frac{1}{n}\sum_{i=1}^n \frac{1}{\sigma_{\theta,-i}^2}\right)^{-1}$. Notice that $w_i(\theta)$ is lower when $\sigma_{\theta,-i}$ is larger, which happens when points are either isolated or lying on the border / envelope of $\mathbb{X}_n$. Equation (1.16) shows that, similarly to the LOO criteria of Section 1.3.2, the GCV criterion can be computed, along with its gradient, in $O(n^3 + dn^2)$ time.

**Kernel alignment** The kernel alignment selection criterion is defined as

$$J_n^{\mathrm{KA}}(\theta) = -\frac{\operatorname{Tr}\left(\mathbf{K}_\theta\mathbf{Z}_n\mathbf{Z}_n^\mathsf{T}\right)}{\|\mathbf{K}_\theta\|_F\|\mathbf{Z}_n\|^2}, \quad (1.17)$$

where $\|\cdot\|_F$ stands for the Frobenius matrix norm. This criterion can be derived from (1.15) by taking $p=-1$ and $q=2$:

$$J_n^{\mathrm{KA}}(\theta) = -\frac{1}{\sqrt{n}\|\mathbf{Z}_n\|^2 J_n^{\mathrm{HL},-1,2}(\theta)}.$$

It was originally proposed in the machine learning literature Cristianini et al. (2001) as a cosine similarity between $\mathbf{K}_\theta$ and the matrix $\mathbf{Z}_n\mathbf{Z}_n^\mathsf{T}$. This criteria is noticeably cheaper than the others, as it does not require to invert $\mathbf{K}_\theta$ and can therefore be computed along with its gradient in $O(dn^2)$ time.

**Remark 7** *We choose to focus in this chapter on three well-known selection criteria (NLL, GCV and KA) that can be seen as special cases of* (1.15)*, corresponding repectively to* $(p,q)$ *equal to* $(1,0)$*,* $(2,-1)$ *and* $(-1,2)$*. The study of new selection criteria, obtained for other values of* $(p,q)$*, is left for future work.*

### 1.3.3 . Hybrid selection criteria

When considering several parameterized models—or, equivalently, when dealing with discrete parameters, such as half-integer values for the regularity parameter of the Matérn covariance—some authors suggest to use one selection criterion to select the parameters in each particular model, and a different one to select the model itself.

For instance, in Jones et al. (1998), the authors select the parameters of a power-exponential covariance function using the NLL selection criterion (i.e., the ML method), and then select a suitable transformation of the output of the simulator, in a finite list of possible choices, using the LOO-SPE criterion. Similarly, the NLL selection criterion is combined in Demay et al. (2021) with a variety of model-validation criteria, including LOO-CRPS and LOO-NLPD.

In Section 1.4 we will denote by NLL/SPE the hybrid method that selects the variance and range parameters of a Matérn covariance function using the NLL criterion, and then minimizes the LOO-SPE criterion to select the regularity parameter $\nu$ in finite list of values.

## 1.4 . Numerical experiments

### 1.4.1 . Methodology

We investigate the problem of parameter selection with an experimental approach consisting of four ingredients: 1) a set of unknown functions $f$ to be predicted using evaluation results on a finite design $\mathbb{X}_n = \{x_1,\ldots,x_n\} \subset \mathbb{X}$; 2) the GP regression method that constructs predictive distributions $P_{\theta,x}$ of $f$ at given $x$s in $\mathbb{X}$, indexed by parameter $\theta$; 3) several selection criteria $J_n$ to choose $\theta$; 4) several criteria to assess the quality of the $P_{\theta,x}$s. Details about each of these ingredients are given below (starting from the last one).

**Criteria to assess the quality of the $P_{\theta,x}$s** A natural way to construct a criterion to assess the quality of the $P_{\theta,x}$s is to choose a scoring rule $S$ and to consider the mean score on a test set $\mathbb{X}_N^{\text{test}} = \{x_1^{\text{test}}, \ldots, x_N^{\text{test}}\} \subset \mathbb{X}$ of size $N$:

$$R(\theta; S) = \frac{1}{N} \sum_{i=1}^{N} S\big(P_{\theta, x_i^{\text{test}}}; f(x_i^{\text{test}})\big)\,. \tag{1.18}$$

**Selection criteria** We shall consider the selection criteria $J_n$ presented in Section 1.3, namely, the LOO-SPE, LOO-NLPD, LOO-CRPS, NLL, GCV, KA and NLL/SPE selection criteria. Given a function $f$ and a design $\mathbb{X}_n$, each selection criterion $J_n$ yields a parameter $\theta_{J_n}$.

**Parameterized GP models** In this work, models are implemented using a custom version of the GPy (Sheffield machine learning group, 2012–2020) Python package (see Supplementary Material, hereafter abbreviated as SM). We assume no observation noise, which corresponds to the interpolation setting. All functions will be centered beforehand to have zero-mean on $\mathbb{X}_N^{\text{test}}$, and we will consider zero-mean GPs only. The anisotropic Matérn covariance function (1.1) is used, with parameter $\theta = (\sigma^2, \rho_1, \ldots, \rho_d, \nu)$, and the regularity parameter $\nu$ is either set a priori to $\nu = \chi + 1/2$, with $\chi \in \{0, 1, 2, 3, 4, d, 2d, \infty\}$, or selected automatically. The latter case will be denoted by $\widehat{\nu}$.

**Remark 8** *Since the covariance matrix of $\mathbf{Z}_n$ can be nearly singular when the range parameters take large values, we define upper bounds on these values in order to avoid the use of* nugget *or* jitter *(see, e.g., Peng and Wu (2014), Ranjan et al. (2011)). Details are provided in the SM.*

**Test functions** The test functions used in the study are described in the next section. They are grouped into collections, and we provide averaged values of mean-score metrics of the form (1.18) for each collection.

### 1.4.2 . Test functions

#### Design of a low-pass filter

Fuhrländer and Schöps Fuhrländer and Schöps (2020) consider the problem of computing, using a frequency-domain PDE solver, the scattering parameters $S_\omega$ of an electronic component called stripline low-pass filter, at several values of the excitation pulsation $\omega$. The geometry of the stripline filter is illustrated on Figure 1.1. It is parameterized using six real valued factors concatenated in a vector $x \in \mathbb{R}^d$, $d = 6$. The objective is to satisfy the low-pass specifications $|S_{2k\pi}(x)| \geq -1\text{dB}$ for $0 \leq k \leq 4$ and $|S_{2k\pi}(x)| \leq -20\text{dB}$ for $5 \leq k \leq 7$. Meeting such requirements is a difficult and time-consuming task.

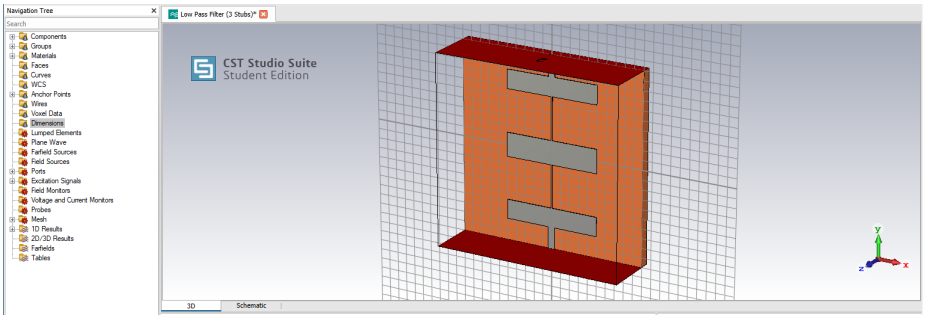



Figure 1.1: A low-pass filter design problem in CST Studio Suite®.

In this chapter we consider the quantities $\mathrm{Re}(S_{2\pi})$, $\mathrm{Re}(S_{6\pi})$, $\mathrm{Re}(S_{10\pi})$ and $\mathrm{Re}(S_{14\pi})$. For three design sizes $n \in \{10d, 20d, 50d\}$, we randomly sample $M = 100$ subsets $\mathbb{X}_n$ of size $n$ from a database of $10000$ simulation results, and use the remaining $N = 10000 - n$ points as test sets. The metric (1.18) is computed and averaged on these $M$ test sets.

### Other test functions

We supplement the above engineering problem with a collection of test functions from the literature. More precisely, we consider the Goldstein-Price function Dixon and Szegö (1978), a one-dimensional version of the Goldstein-Price function (see SM for details), the Mystery function Martin and Simpson (2003), the Borehole function Worley (1987), several collections obtained from the GKLS simulator Gaviano et al. (2003), and the rotated Rosenbrock collection from the BBOB benchmark suite Hansen et al. (2009).

The GKLS simulator has a "smoothness" parameter $k \in \{0, 1, 2\}$ controlling the presence of non-differentiabilities on some nonlinear subspaces—the trajectories being otherwise infinitely differentiable. For both GKLS and Rosenbrock, two different values of the input dimension were considered ($d = 2$ and $d = 5$). The resulting set of twelve problems—considering that changing the value of $k$ or $d$ defines a new problem—is summarized in Table 1.3.

For each problem, we consider three design sizes $n \in \{10d, 20d, 50d\}$. For the GKLS and Rosenbrock collections, we directly used the collections of test functions provided by the authors ($M = 100$ and 15 functions, respectively). For a given dimension, they are all evaluated on the same space-filling designs $\mathbb{X}_n$. For each of the remaining problems, we used a single test function, evaluated on $M = 100$ random space-filling designs $\mathbb{X}_n$, thereby creating collections of 100 data sets. In both cases, maximin (over 1000 repetitions) Latin hypercube designs (see, e.g., McKay et al., 2000) are used.

A Sobol' sequence $\mathbb{X}_N^{\text{test}}$ of size $N = 10000$ is used as test set and the functions

Table 1.3: Twelve benchmark problems

| Problem | Goldstein-Price | Mystery | $\mathrm{GKLS}_{k=0}$ | $\mathrm{GKLS}_{k=1}$ | $\mathrm{GKLS}_{k=2}$ | Rosenbrock | Borehole |
|---|---|---|---|---|---|---|---|
| $d$ | $\{1,2\}$ | 2 | $\{2,5\}$ | $\{2,5\}$ | $\{2,5\}$ | $\{2,5\}$ | 8 |

Table 1.4: Averages (over the $M = 100$ designs) of the $R(\theta; S^{\mathrm{SPE}})$ values for $\mathrm{Re}(S_{6\pi})$ with $n = 10d = 60$. Optimal $R^\star$ values are given in the rightmost column for comparison. For comparison also, $R(\theta; S^{\mathrm{SPE}}) = 3.26 \cdot 10^{-4}$ for the $J_n^{\mathrm{NLL/SPE}}$ selection criterion, which also selects $\nu$ (see Section 1.3.3). The gray scale highlights the order of magnitude of the discrepancies.

| Scoring rule: $S^{\mathrm{SPE}}$ | NLL | LOO-SPE | LOO-NLPD | LOO-CRPS | KA | GCV | $R^\star$ |
|---|---|---|---|---|---|---|---|
| $\nu = 1/2$ | $4.94 \cdot 10^{-2}$ | $5.11 \cdot 10^{-2}$ | $4.84 \cdot 10^{-2}$ | $4.84 \cdot 10^{-2}$ | $4.29 \cdot 10^{-1}$ | $4.73 \cdot 10^{-2}$ | $4.44 \cdot 10^{-2}$ |
| $\nu = 3/2$ | $3.85 \cdot 10^{-3}$ | $4.24 \cdot 10^{-3}$ | $3.52 \cdot 10^{-3}$ | $3.59 \cdot 10^{-3}$ | $3.82 \cdot 10^{-1}$ | $3.45 \cdot 10^{-3}$ | $2.97 \cdot 10^{-3}$ |
| $\nu = 5/2$ | $4.02 \cdot 10^{-4}$ | $5.11 \cdot 10^{-4}$ | $4.18 \cdot 10^{-4}$ | $4.31 \cdot 10^{-4}$ | $3.71 \cdot 10^{-1}$ | $4.93 \cdot 10^{-4}$ | $3.21 \cdot 10^{-4}$ |
| $\nu = 7/2$ | $2.88 \cdot 10^{-4}$ | $4.33 \cdot 10^{-4}$ | $3.73 \cdot 10^{-4}$ | $3.86 \cdot 10^{-4}$ | $3.54 \cdot 10^{-1}$ | $4.75 \cdot 10^{-4}$ | $2.26 \cdot 10^{-4}$ |
| $\nu = 9/2$ | $2.96 \cdot 10^{-4}$ | $4.39 \cdot 10^{-4}$ | $4.22 \cdot 10^{-4}$ | $3.95 \cdot 10^{-4}$ | $3.42 \cdot 10^{-1}$ | $5.44 \cdot 10^{-4}$ | $2.26 \cdot 10^{-4}$ |
| $\nu = 13/2$ | $3.15 \cdot 10^{-4}$ | $4.80 \cdot 10^{-4}$ | $4.48 \cdot 10^{-4}$ | $4.25 \cdot 10^{-4}$ | $3.29 \cdot 10^{-1}$ | $6.43 \cdot 10^{-4}$ | $2.32 \cdot 10^{-4}$ |
| $\nu = 25/2$ | $3.46 \cdot 10^{-4}$ | $5.29 \cdot 10^{-4}$ | $4.92 \cdot 10^{-4}$ | $4.61 \cdot 10^{-4}$ | $3.14 \cdot 10^{-1}$ | $7.43 \cdot 10^{-4}$ | $2.45 \cdot 10^{-4}$ |
| $\nu = \infty$ | $3.80 \cdot 10^{-4}$ | $6.34 \cdot 10^{-4}$ | $5.38 \cdot 10^{-4}$ | $5.38 \cdot 10^{-4}$ | $2.94 \cdot 10^{-1}$ | $8.75 \cdot 10^{-4}$ | $2.60 \cdot 10^{-4}$ |
| $\nu \in \{1/2, \cdots, \infty\}$ | $3.07 \cdot 10^{-4}$ | $4.90 \cdot 10^{-4}$ | $4.64 \cdot 10^{-4}$ | $4.31 \cdot 10^{-4}$ | $2.98 \cdot 10^{-1}$ | $6.89 \cdot 10^{-4}$ | $2.13 \cdot 10^{-4}$ |

are centered and normalized to unit variance on these test sets.

### 1.4.3 . Results and findings

## A close look at one of the problems

Tables 1.4 and 1.5 provide a detailed view of the results obtained on one of the test problems—namely, the output $f = \mathrm{Re}(S_{6\pi})$ with $n = 10d = 60$ of the low-pass filter case (see Section 1.4.2).

The results presented in these tables are the scores $R(\theta; S)$, averaged over the $M = 100$ random instances of the problem, where $\theta$ is selected using different selection criteria (along columns), and the regularity of the Matérn covariance varies or is selected automatically (along rows). The scoring rule for assessing the quality of the predictions is the SPE in Table 1.4 and the IS at level 95% in Table 1.5. (A similar table, not shown here, is presented in the SM for the CRPS.)

For comparison, Table 1.4 also provides the optimal values $R^\star$ obtained by direct minimization of the score (1.18). They can be used to assess the loss of predictive accuracy of the selected models, which are constructed using a limited number of observations, with respect to the best model that could have been obtained if the test data had also been used to select the parameters.

Table 1.4 and Table 1.5 support the fact that, for this particular problem, the NLL and NLL/SPE criteria are the best choices for selecting $\theta$ in terms of the

Table 1.5: Same as Table 1.4 but for averages of $R(\theta; S^{\mathrm{IS}}_{0.95})$. Using $J_n^{\mathrm{NLL/SPE}}$ gives $R(\theta; S^{\mathrm{IS}}_{0.95}) = 7.06 \cdot 10^{-2}$.

| Scoring rule: $S^{\mathrm{IS}}_{0.95}$ | NLL | LOO-SPE | LOO-NLPD | LOO-CRPS | KA | GCV |
|---|---|---|---|---|---|---|
| $\nu = 1/2$ | $1.44 \cdot 10^{0}$ | $1.38 \cdot 10^{0}$ | $1.34 \cdot 10^{0}$ | $1.48 \cdot 10^{0}$ | $3.69 \cdot 10^{0}$ | $1.32 \cdot 10^{0}$ |
| $\nu = 3/2$ | $2.80 \cdot 10^{-1}$ | $3.05 \cdot 10^{-1}$ | $2.75 \cdot 10^{-1}$ | $2.88 \cdot 10^{-1}$ | $3.43 \cdot 10^{0}$ | $2.69 \cdot 10^{-1}$ |
| $\nu = 5/2$ | $9.30 \cdot 10^{-2}$ | $9.42 \cdot 10^{-2}$ | $8.55 \cdot 10^{-2}$ | $9.11 \cdot 10^{-2}$ | $3.36 \cdot 10^{0}$ | $9.18 \cdot 10^{-2}$ |
| $\nu = 7/2$ | $6.82 \cdot 10^{-2}$ | $8.82 \cdot 10^{-2}$ | $8.42 \cdot 10^{-2}$ | $9.10 \cdot 10^{-2}$ | $3.23 \cdot 10^{0}$ | $9.17 \cdot 10^{-2}$ |
| $\nu = 9/2$ | $6.50 \cdot 10^{-2}$ | $9.08 \cdot 10^{-2}$ | $9.30 \cdot 10^{-2}$ | $9.53 \cdot 10^{-2}$ | $3.14 \cdot 10^{0}$ | $1.00 \cdot 10^{-1}$ |
| $\nu = 13/2$ | $6.48 \cdot 10^{-2}$ | $9.95 \cdot 10^{-2}$ | $9.99 \cdot 10^{-2}$ | $1.03 \cdot 10^{-1}$ | $3.02 \cdot 10^{0}$ | $1.16 \cdot 10^{-1}$ |
| $\nu = 25/2$ | $6.77 \cdot 10^{-2}$ | $1.10 \cdot 10^{-1}$ | $1.08 \cdot 10^{-1}$ | $1.12 \cdot 10^{-1}$ | $2.90 \cdot 10^{0}$ | $1.29 \cdot 10^{-1}$ |
| $\nu = \infty$ | $7.23 \cdot 10^{-2}$ | $1.23 \cdot 10^{-1}$ | $1.16 \cdot 10^{-1}$ | $1.23 \cdot 10^{-1}$ | $2.74 \cdot 10^{0}$ | $1.44 \cdot 10^{-1}$ |
| $\nu \in \{1/2, \cdots, \infty\}$ | $6.70 \cdot 10^{-2}$ | $1.04 \cdot 10^{-1}$ | $1.06 \cdot 10^{-1}$ | $1.08 \cdot 10^{-1}$ | $2.78 \cdot 10^{0}$ | $1.23 \cdot 10^{-1}$ |

SPE and the IS scores, both for a prescribed regularity $\nu$ and when $\nu$ is selected automatically (the NLL/SPE being only available for the latter case). Except for the KA criterion, however, the other selection criteria are never very far behind. Elements provided as SM show similar findings using the CRPS validation score.

Strikingly enough, for both scoring rules, the variations of the average score are much larger when the model changes than when the selection criterion changes. If a Matérn covariance function with fixed regularity is used, as is often done in practice, then the best results are obtained for all criteria (except KA) when $\nu$ takes the values $7/2$, $9/2$ and $13/2$. The values of $R^*$ (Table 1.4) confirm that these are indeed the best fixed-$\nu$ models on this problem for the SPE score. Since these optimal values were not known beforehand, it is a relief to see (cf. last row of each table) that comparable performances can be achieved on this problem by selecting $\nu$ automatically.

## Statistical analysis of the benchmark results

Tables similar to Tables 1.4 and 1.5 have been produced for all the $(4+12) \times 3 = 48$ test problems presented in Section 1.4.2, for the three scoring rules (SPE, CRPS and IS). We present in this section some graphical summaries and statistical analyses of these results. The individual tables for each problem are provided in the SM.

**Remark 9** *The poor performance of the KA criterion, already observed in Tables 1.4 and 1.5, is confirmed by the results (not shown) on all the other problems. We conclude that this selection criterion should not be used in practice, and exclude it from the analyses of this section in order to refine the comparison between the remaining ones.*

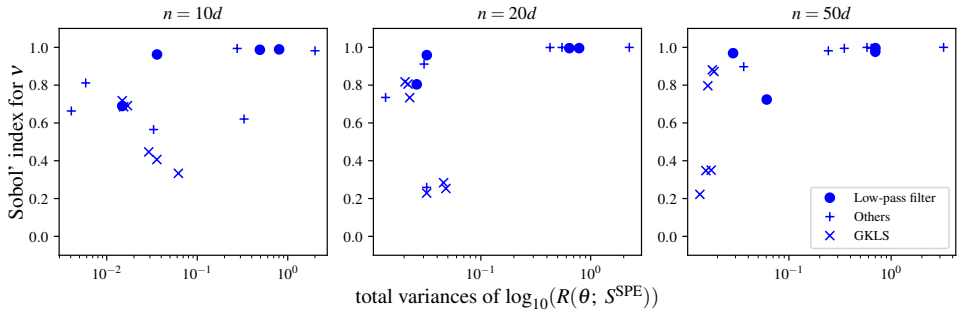



Figure 1.2: Parts of total variances of $\log_{10}(R)$s explained by $\nu$ for $S^{\mathrm{SPE}}$ using a one-factor ANOVA. Each point represents the variations of $\log_{10}(R)$ for one of the 16 problems from Section 1.4.2, split by design size, with KA and GCV excluded. The model explains almost all the variations for problems that exhibit significant fluctuations of $\log_{10}(R)$ (at the right of the figure).

**Sensitivity analysis** We observed in Section 1.4.3 that the choice of the model—more specifically, of the regularity parameter of the Matérn covariance function—was more important than that of a particular selection criterion (excluding KA of course). To confirm this finding, a global sensitivity analysis of the logarithm of the average score, where the average is taken over the $M = 100$ instances of each problem, has been performed on each problem. The average score, for a given scoring rule, depends on two discrete factors: the selection criterion and the regularity $\nu$ of the Matérn covariance function. We present on Figure 1.2, for the SPE scoring rule, the Sobol' sensitivity index for the latter factor as a function of the total variance. Observe that, for the problems where the total variance is large, the Sobol index is typically very close to one, which indicates that the variability is indeed mainly explained by the choice of model. Similar conclusions hold for the other scoring rules (results not shown, see SM).

**Comparison of the covariance models** Figure 1.3 compares the average values of $R(\theta;S^{\mathrm{SPE}})$ when ML is used on the set of GKLS problems, which have low regularities, and on the set of low-pass filter problems, which contains very smooth instances.

Observe first that the fixed-$\nu$ models rank differently on these two sets of problems, as expected considering the actual regularity of the underlying functions: low values of $\nu$ perform better on the GKLS problems and worse on the low-pass filter case. Furthermore, it appears that underestimating the regularity (on the low-pass filter case) has much more severe consequences than overestimating it (on the GKLS problems) according to the SPE score, as suggested by the theoretical results of Stein (1999), Narcowich et al. (2006)—see (Scheurer et al., 2013, Section 6)

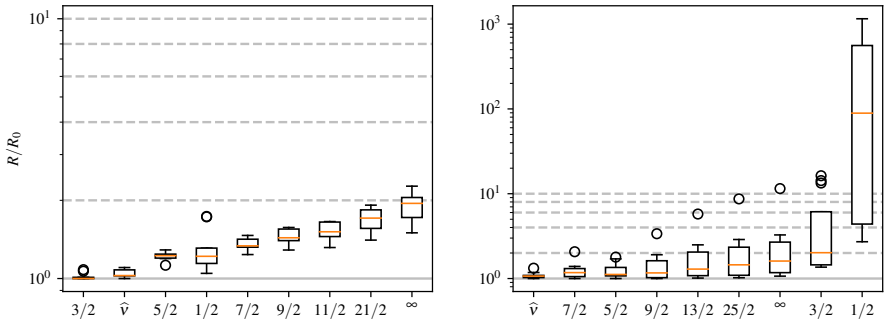



Figure 1.3: Box plots of $R/R_0$ using $J_n^{\mathrm{NLL}}$ as selection criterion and $S^{\mathrm{SPE}}$ as quality assessment criterion, for different choices of regularity. Here, $R_0$ stands for the best value of $R$ on each problem (among all models). Left: All $5d$ GKLS problems. Right: All low-pass filter problems. The box plots are sorted according to their upper whisker. Grey dashed lines: $R/R_0 = 2,4,6,8,10$.

for a discussion—and Tuo and Wang (2020).

Another important conclusion from Figure 1.3 is that very good results can be obtained by selecting the regularity parameter $\nu$ automatically, jointly with the other parameters (using the NLL criterion in this case). On the GKLS problems, the results with selected $\nu$ are not far from those of the best fixed-$\nu$ model under consideration ($\nu = 3/2$); in the low-pass filter case, they are even better than those obtained with the best fixed-$\nu$ models ($\nu = 5/2$ or $7/2$). In other words, the regularity needs not be known in advance to achieve good performances, which is a very welcome practical result. This conclusion is also supported, for NLL, by the additional results provided in the SM for the other problems and for the three scoring rules.

Concerning the other selection criteria the situation is more contrasted (see SM): the automatic selection of $\nu$ using these criteria still performs very well for smooth problems, but not always, in particular with GCV, for the less regular problems of the GKLS class. This is especially true when the sample size is small ($n = 10d$).

**Comparison of the selection criteria** Figure 1.4 compares the distributions of the average values of the SPE and IS scores for all selection criteria (except KA) on all test instances, in the case of a Matérn covariance function with automatically selected regularity. As a preliminary observation, note that for most cases the ratio $R/R_0$ remains under two (first horizontal dashed line), which confirms that the differences between selection criteria are much milder than those between covariance models (recall Figure 1.3).

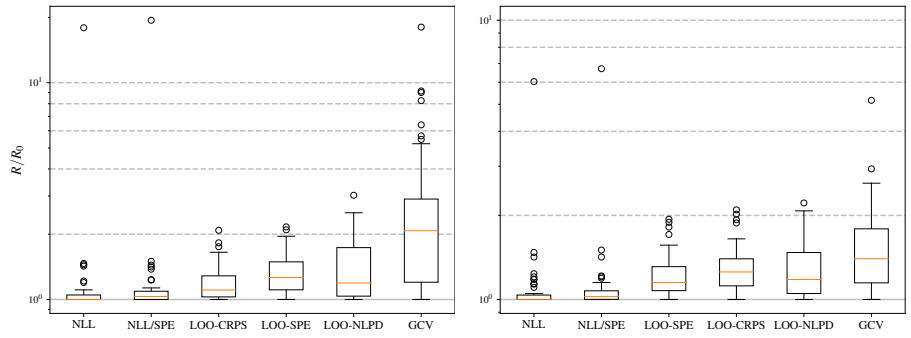



Figure 1.4: Box plots of $R/R_0$ for different selection criteria. Each box plot is constructed using all problems, with automatically selected $\nu$, and $R_0$ stands for the best value of $R$ on each problem (among all selection criteria). The left (resp. right) panel uses the $S^{\mathrm{SPE}}$ (resp. $S^{\mathrm{IS}}_{0.95}$) as quality assessment criterion. Sorting of box plots and horizontal lines: as in Figure 1.3.

A closer look at Figure 1.4 reveals that the rankings of criteria obtained for both scoring rules are almost identical. The ranking for the CRPS scoring rule (not shown) is the same as the one for SPE. GCV provides the worst performance for all scoring rules, followed by LOO-NLPD, while NLL dominates the ranking (for all scoring rules as well).

**Remark 10** *Observe on Figure 1.4 that LOO-SPE is surprisingly significantly less accurate than NLL according to $S^{\mathrm{SPE}}$. More generally, choosing a scoring rule $S$ for the LOO criterion does not guarantee the highest precision according to this particular score.*

**Robustness** LOO-SPE is commonly claimed in the literature (see, notably, Bachoc (2013b)) to provide a certain degree of robustness with respect to model misspecification. According to this claim, LOO-SPE would be expected to somehow mitigate the loss of predictive accuracy with respect to likelihood-based approaches incurred by an ill-advised choice of covariance function. Our detailed results (see SM) suggest that this effect indeed exists when the regularity is severely under-estimated (e.g., $\nu = 1/2$ for the low-pass filter problems), but is actually quite small, and should not be used to motivate the practice of setting $\nu$ to an arbitrary value. A similar effect exists for LOO-CRPS, LOO-NLPD and GCV as well. Quite surprisingly, NLL turns out to be more robust than LOO-SPE (and the other criteria) in the case of over-smoothing.

## 1.5 . Conclusions

A large variety of selection criteria for Gaussian process models is available from the literature, with little theoretical or empirical guidance on how to choose the right one for applications. Our benchmark study with the Matérn family of covariance functions in the noiseless (interpolation) case indicates that the NLL selection criterion—in other words, the ML method—provides performances that are, in most situations and for all the scoring rules that were considered (SPE, CRPS and IS at 95%), better than or comparable to those of the other criteria. Considering that all the criteria tested in the study (except KA) have a similar computational complexity, this provides a strong empirical support to the ML method—which is already the *de facto* standard in most statistical software packages implementing Gaussian process interpolation.

Another important lesson learned from our benchmark study is that the choice of the family of models, and in particular of the family of covariance functions, has very often a bigger impact on performance than that of the selection criterion itself. This is especially striking when the actual function is smooth, and very irregular covariance function such as the Matérn covariance with regularity $1/2$ is used to perform Gaussian process interpolation. In such a situation, NLL is actually outperformed by other criteria such as LOO-SPE or LOO-CRPS, which thus appear to be more "robust to model misspecification". However, the small gain of performance, which is achieved by using LOO-SPE or LOO-CRPS instead of NLL in this case, is generally negligible with respect to the much larger loss induced by choosing an inappropriate covariance function in the first place.

Our final recommendation, supported by the results of the benchmark, is therefore to select, if possible, the regularity of the covariance function automatically, jointly with the other parameters, using the NLL criterion. A minimal list of candidate values for the regularity parameter should typically include $1/2$, $3/2$, $5/2$, $7/2$ and $+\infty$ (the Gaussian covariance function). Should a situation arise where a default value of $\nu$ is nevertheless needed, our recommendation would be to choose a reasonably large value such as $\nu = 7/2$, since under-smoothing has been seen to have much more severe consequences than over-smoothing. More generally, our numerical results support the fact that choosing a model carefully is important, and probably so not only in the class of Matérn covariance functions.

However, it should be kept in mind that the study focuses on cases where the number of parameters is small with respect to the number of observations (in particular, we considered zero-mean GPs with an anisotropic stationary Matérn covariance function, which have $d+2$ parameters, and we took care of having $n \gg d$). When $d$ is large, or when the number of parameters increases, it seems to us that other selection criteria should be considered, and that the introduction of regularization terms would be required.

For future work, it would be very interesting to consider the performance of using selection criteria against a fully Bayesian approach. Another direction would be to extend this study to the case of regression, which is also used in many

applications, when dealing with stochastic simulators, for instance.

## 1.6 . Additional Material: a numerical study about the choice of $\nu$ for Bayesian optimization

### 1.6.1 . Related works

We now complete the numerical experiments of Section 1.4 by an optimization benchmark. We refer to Section 2.1 for a brief description and references about the Efficient Global Optimization (EGO) algorithm and the Expected Improvement (EI) sampling criterion.

Recently, Le Riche and Picheny (2021) made an insightful empirical comparison of some EGO variants using the `COCO` (COmparing Continuous Optimizers, (Hansen et al., 2021)) benchmark. The authors point out that most of the available benchmarks in the literature (see the provided references, to which we can add the recent work of Merrill et al. (2021)) deal with the sampling criterion or Bayesian optimization variants. Therefore they choose to stick to the EI and investigate other levers with, for instance, the choice of the mean and the covariance function families, the size of the initial design, and some input or output transformations. They treated the case of the covariance function by testing EGO with the Matérn subfamilies corresponding to $\nu = 1/2$ and $\nu = 5/2$ and conclude that the best choice depends on the smoothness of the function to be optimized; with the $\nu = 5/2$ subfamily being more often preferable on the `COCO` benchmark.

### 1.6.2 . Methodology and test cases

Given the findings from Section 1.4, we restrict ourselves by sticking to the NLL criterion and investigating the choice of the regularity parameter of a Matérn covariance function for Bayesian optimization in the wake of Le Riche and Picheny (2021). We proceed by considering a benchmark inspired by (Feliot, 2017, Section 2.5.3) and described in Table 1.6.

For this benchmark, we also stick to the classical EI with: 1) independent models for the outputs; 2) no input or output deformation; 3) a constant trend function; 4) the optimization of the sampling criterion conducted with an SMC approach (Feliot et al., 2017) that uses the PICPI density (the probability of improvement with a relaxation on the constraints, see Feliot, 2017, Section 3.2.4); space-filling design of experiments of size $3d$; and 5) the Matérn subfamilies corresponding to the fixed values $\nu = 1/2, 3/2, 5/2, 7/2, 9/2, 13/2, 17/2, 21/2, \infty$ and their union, i.e., with $\nu$ being selected automatically from data. Observe that the regularity parameters of the outputs are untied only when $\nu$ is automatically selected.

For each test case, we run the optimization until the prescribed target from Table 1.6 is achieved or a maximum budget of $300$ evaluations—regardless of the dimension—is reached.

### 1.6.3 . Results

Table 1.6: Optimization benchmark composed of: 1) the Goldstein-Price function Dixon and Szegö (1978) along with a log version as in Part III; and 2) the constrained mono-objective cases from (Feliot, 2017, Section 2.5.3). Feliot (2017) considers several versions of the "g10" problem. In our benchmark, we will consider: 1) the g10-ARH version (see the "g10" problem from Hedar and Ahmed, 2004); 2) the g10-RR version (see the "g10" problem from Regis, 2014); and 3) the g10-PF version (see the "modified-g10" problem from Feliot et al., 2017). Some characteristics and "target" values for the problems—mostly taken from (Feliot, 2017, Table 2.1)—are provided.

| Pbm | $d$ | $q$ | $\Gamma(\%)$ | Target |
|---|---|---|---|---|
| Goldstein-Price | 2 | 0 | 100 | 3.001 |
| Goldstein-Price Log | 2 | 0 | 100 | $\log(3.001)$ |
| g1 | 13 | 9 | $4 \cdot 10^{-4}$ | $-14.85$ |
| g3mod | 20 | 1 | $10^{-4}$ | $-0.33$ |
| g5mod | 4 | 5 | $8.7 \cdot 10^{-2}$ | 5150 |
| g6 | 2 | 2 | $6.6 \cdot 10^{-3}$ | $-6800$ |
| g7 | 10 | 8 | $10^{-4}$ | 25 |
| g8 | 2 | 2 | 0.86 | $-0.09$ |
| g9 | 7 | 4 | 0.52 | 1000 |
| g10-ARH | 8 | 6 | $7 \cdot 10^{-4}$ | 8000 |
| g10-RR | 8 | 6 | $7 \cdot 10^{-4}$ | 8000 |
| g10-PF | 8 | 6 | $7 \cdot 10^{-4}$ | 8000 |
| g13-mod | 5 | 3 | 4.5 | 0.005 |
| g16 | 5 | 38 | $1.3 \cdot 10^{-2}$ | $-1.8$ |
| g18 | 9 | 13 | $2 \cdot 10^{-10}$ | $-0.8$ |
| g19 | 15 | 5 | $3.4 \cdot 10^{-3}$ | 40 |
| g24 | 2 | 2 | 44.3 | $-5$ |
| SR7 | 7 | 11 | $9.3 \cdot 10^{-2}$ | 2995 |
| PVD4 | 4 | 3 | $5.6 \cdot 10^{-1}$ | 6000 |
| WB4 | 4 | 6 | $5.6 \cdot 10^{-2}$ | 2.5 |

The results are shown in Table 1.7 and Table 1.8 and show significant variations with respect to $\nu$. The smoother the covariance function, the easier it is to solve g1, g5mod, g7, g9, g10-ARH, g16, g18, g19, g24, WB4, and Goldstein-Price (with the traditional $\nu = 5/2$ value underperforming significantly on g9, g19, WB4 and Goldstein-Price), whereas g8, g10-RR, g10-PF, g13mod, SR7, and Goldstein-Price (Log) require covariance functions that are not too smooth. For the remainings, the cases g3mod, g6, and g10-RR remain unsolved[1], and the case PVD4 cannot be solved by any fixed $\nu$ value.

The $\nu = 1/2$ covariance function family yields the most dramatic underperformances, but improves the percentage of successful runs from $0\%$ to $10\%$ on g10-RR. Although the problem remains unsolved most of the time, taking $\nu = 1/2$—or selecting it automatically—yields an average best feasible score of about $8400$ against $9000$ for the other families, which is closer to the target from Table 1.6. Rough covariance functions are more convenient to solve this problem because some of its outputs look like discontinuous functions, as shown in Figure 1.5.

Overall, selecting $\nu$ automatically always leads to being very close to the top performer except maybe for g13mod and SR7, where it shows only moderate underperformance. Observe that it outperforms every fixed regularity on Goldstein-Price and PVD4. The PVD4 problem is instructive since it is a situation where it is preferable to keep the regularity parameters untied. Regarding Goldstein-Price, a closer inspection of the numerical results shows that the selected regularity increases progressively during the optimization, which may be a possible explanation that it seems to provide both the robustness of the moderate regularities and the speed of the larger ones.

### 1.6.4 . Conclusion

This optimization benchmark study provides additional support to the fact that the regularity parameter of a Matérn covariance function has an important contribution to the predictivity of Gaussian process interpolation models. In this section, the predictivity was measured by the performances of corresponding Bayesian optimization algorithms. As in Section 1.4, different regularity parameters can lead to drastically different optimization results. The more dramatic underperformances are obtained using too rough covariance functions, and the automatic selection of the regularity parameter is a satisfactory adaptive methodology. Furthermore,

[1] Case g6 is cheaply solved by (Feliot, 2017). The reason is that the two constraints are quadratic functions that are easily modeled by any covariance function that is at least twice differentiable. However, the problem is two-dimensional, and the volume of the feasible space is of order $10^{-5}$, so either the optimizer succeeds rapidly, or it crashes for having evaluations that are too close in this region without having found the solution. The success of Feliot (2017) may be explained by his special extended domination rule and the fact that the experiments were conducted using the STK (Bect et al., 2011–2021) toolbox, which is more mature than our refined version of GPy.

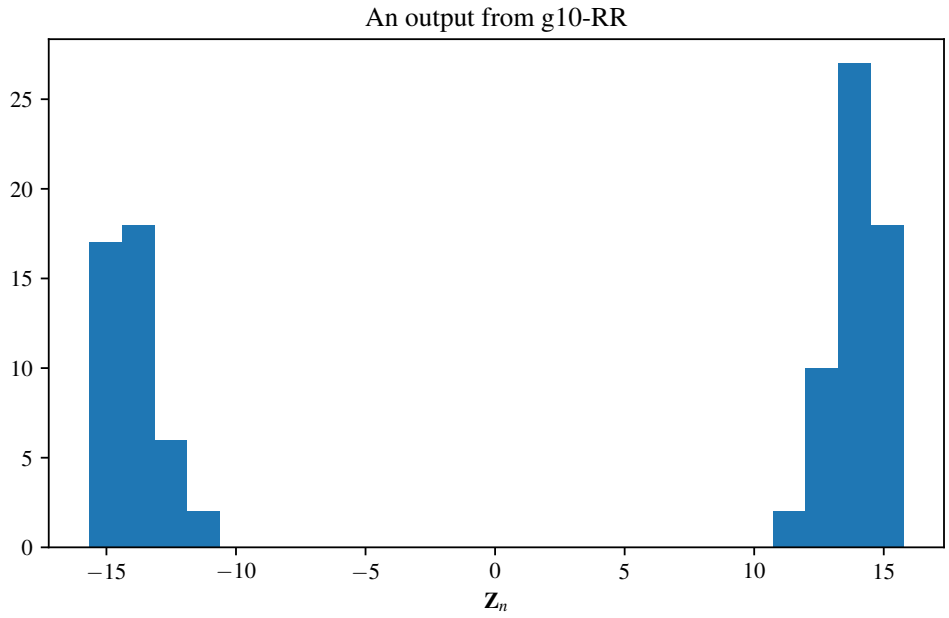



Figure 1.5: A histogram for the constraint $c_6 \leq 0$ from the g10-RR problem for a space-filling design of experiments of $n = 60$ points. Observe that, although the function is infinitely differentiable, its steep variations make it looks like a piecewise constant function.

Table 1.7: A small unconstrained mono-objective optimization benchmark. We show between parenthesis the fraction of the 30 runs that manage to reach a given target value in 300 evaluations, and the average number of evaluations to reach it is also provided (with unsuccessful runs counted as 300). The best value in a column is in bold and a red cell indicates a fraction of successful runs below 66%. Moreover, the grayscale highlights the variations with respect to the best value for a given column.

| | Goldstein-Price | Goldstein-Price Log |
|---|---|---|
| $\nu = 1/2$ | $+\infty(0.0)$ | $207.87(0.4)$ |
| $\nu = 3/2$ | $+\infty(0.0)$ | $\mathbf{47.23(1.0)}$ |
| $\nu = 5/2$ | $238.27(0.67)$ | $76.17(1.0)$ |
| $\nu = 7/2$ | $134.53(1.0)$ | $100.47(1.0)$ |
| $\nu = 9/2$ | $100.27(1.0)$ | $92.0(1.0)$ |
| $\nu = 13/2$ | $101.2(0.93)$ | $123.13(0.87)$ |
| $\nu = 17/2$ | $118.13(0.83)$ | $112.97(0.9)$ |
| $\nu = 21/2$ | $92.0(0.93)$ | $128.2(0.83)$ |
| $\nu = \infty$ | $112.93(0.83)$ | $122.2(1.0)$ |
| $\nu \in \{1/2, \cdots, \infty\}$ | $\mathbf{83.6(1.0)}$ | $53.57(1.0)$ |

another situation was identified: the case of optimization problems with several outputs (an objective and one or several constraints), where using different regularity values for the outputs leads to very significant benefits compared to using any single value.

It is worth noting also that Le Riche and Picheny (2021) resort to `DiceOptim` (Roustant et al., 2012b), which uses *tensorized* covariance functions instead of the *isotropic* ones (1.1). These two ways of building multivariate covariance functions are discussed by (Stein, 2005b, 1999, Section 2.11), who provide theoretical arguments showing that they lead to very different models. Although Stein (2005b) disqualifies tensorized covariance functions for geostatistical applications, the situation is surprisingly unclear for other fields with no clear material or guidelines on this particular topic.

Table 1.8: A small constrained mono-objective optimization benchmark. The results are presented the same way as in Table 1.7, but using the number of iterations to find a feasible point that reach the target.

| | g1 | g3mod | g5mod | g6 | g7 | g8 |
|---|---|---|---|---|---|---|
| $\nu = 1/2$ | $+\infty$(0.0) | $+\infty$(0.0) | 141.23(1.0) | $+\infty$(0.0) | $+\infty$(0.0) | 106.23(0.87) |
| $\nu = 3/2$ | 67.4(0.97) | $+\infty$(0.0) | 17.03(1.0) | $+\infty$(0.0) | 97.1(1.0) | 32.97(1.0) |
| $\nu = 5/2$ | 56.57(1.0) | $+\infty$(0.0) | 16.4(1.0) | $+\infty$(0.0) | 52.6(1.0) | 27.9(1.0) |
| $\nu = 7/2$ | 56.9(1.0) | $+\infty$(0.0) | **16.1(1.0)** | $+\infty$(0.0) | 50.07(1.0) | 28.9(1.0) |
| $\nu = 9/2$ | 64.6(0.97) | $+\infty$(0.0) | 16.17(1.0) | $+\infty$(0.0) | 50.53(1.0) | 28.4(1.0) |
| $\nu = 13/2$ | 56.67(1.0) | $+\infty$(0.0) | 16.13(1.0) | $+\infty$(0.0) | 50.63(1.0) | 29.03(1.0) |
| $\nu = 17/2$ | 56.03(1.0) | $+\infty$(0.0) | 16.53(1.0) | $+\infty$(0.0) | 50.87(1.0) | 38.07(1.0) |
| $\nu = 21/2$ | 56.87(1.0) | $+\infty$(0.0) | 16.17(1.0) | $+\infty$(0.0) | 50.4(1.0) | 36.17(1.0) |
| $\nu = \infty$ | 64.67(0.97) | $+\infty$(0.0) | 16.33(1.0) | $+\infty$(0.0) | **49.8(1.0)** | 31.17(1.0) |
| $\nu \in \{1/2, \cdots, \infty\}$ | **54.83(1.0)** | $+\infty$(0.0) | 16.17(1.0) | $+\infty$(0.0) | 50.13(1.0) | **27.37(1.0)** |

| | g9 | g10-ARH | g10-RR | g10-PF | g13mod | g16 |
|---|---|---|---|---|---|---|
| $\nu = 1/2$ | 298.4(0.03) | 260.27(0.57) | 291.33(0.1) | 120.2(1.0) | $+\infty$(0.0) | 218.1(0.63) |
| $\nu = 3/2$ | 98.07(1.0) | 36.93(1.0) | $+\infty$(0.0) | **54.37(1.0)** | 184.63(0.93) | 43.63(1.0) |
| $\nu = 5/2$ | 52.1(1.0) | 27.83(1.0) | $+\infty$(0.0) | 62.73(1.0) | 169.47(0.93) | 33.77(1.0) |
| $\nu = 7/2$ | 45.97(1.0) | **27.73(1.0)** | $+\infty$(0.0) | 67.33(1.0) | **155.17(1.0)** | 30.77(1.0) |
| $\nu = 9/2$ | **43.1(1.0)** | 27.93(1.0) | $+\infty$(0.0) | 75.73(1.0) | 170.73(0.87) | 28.73(1.0) |
| $\nu = 13/2$ | 45.87(1.0) | 27.93(1.0) | $+\infty$(0.0) | 74.27(1.0) | 224.87(0.57) | 28.13(1.0) |
| $\nu = 17/2$ | 45.93(1.0) | 27.9(1.0) | $+\infty$(0.0) | 76.93(1.0) | 235.5(0.43) | 31.43(1.0) |
| $\nu = 21/2$ | 43.43(1.0) | 28.13(1.0) | $+\infty$(0.0) | 82.5(1.0) | 239.33(0.53) | 28.5(1.0) |
| $\nu = \infty$ | 47.3(1.0) | 27.97(1.0) | $+\infty$(0.0) | 82.57(1.0) | 232.63(0.5) | 29.87(1.0) |
| $\nu \in \{1/2, \cdots, \infty\}$ | 43.63(1.0) | 27.83(1.0) | 297.27(0.1) | 54.4(1.0) | 191.53(0.8) | **26.7(1.0)** |

| | g18 | g19 | g24 | PVD4 | SR7 | WB4 |
|---|---|---|---|---|---|---|
| $\nu = 1/2$ | $+\infty$(0.0) | $+\infty$(0.0) | 24.23(1.0) | 246.0(0.37) | $+\infty$(0.0) | 300.53(0.03) |
| $\nu = 3/2$ | 187.13(0.87) | $+\infty$(0.0) | 11.77(1.0) | 286.1(0.2) | **49.07(1.0)** | 67.27(1.0) |
| $\nu = 5/2$ | 59.5(1.0) | 104.2(1.0) | 11.0(1.0) | 262.97(0.27) | 58.67(1.0) | 47.3(1.0) |
| $\nu = 7/2$ | 54.9(1.0) | 84.97(1.0) | 10.63(1.0) | 274.8(0.13) | 57.43(1.0) | 42.37(1.0) |
| $\nu = 9/2$ | 54.3(1.0) | 84.23(1.0) | 10.6(1.0) | 279.07(0.23) | 57.87(1.0) | 40.8(1.0) |
| $\nu = 13/2$ | 53.63(1.0) | **83.27(1.0)** | 10.47(1.0) | 287.9(0.13) | 57.93(1.0) | **40.4(1.0)** |
| $\nu = 17/2$ | 53.83(1.0) | 84.6(1.0) | 10.43(1.0) | 286.37(0.1) | 54.0(1.0) | 41.03(1.0) |
| $\nu = 21/2$ | 53.0(1.0) | 84.07(1.0) | 10.5(1.0) | 289.37(0.13) | 53.5(1.0) | 40.53(1.0) |
| $\nu = \infty$ | **52.87(1.0)** | 83.37(1.0) | 10.47(1.0) | 289.4(0.07) | 56.73(1.0) | 43.03(1.0) |
| $\nu \in \{1/2, \cdots, \infty\}$ | 55.57(1.0) | 84.83(1.0) | **10.23(1.0)** | **195.67(0.73)** | 59.2(1.0) | 41.43(1.0) |

# 2 - Numerical issues: the case of maximum-likelihood

This chapter is a reproduction of Basak et al. (2021) with few modifications.

S. Basak, S. J. Petit, J. Bect, and E. Vazquez. Numerical issues in maximum likelihood parameter estimation for Gaussian process regression. In *7th International Conference on machine Learning, Optimization and Data science*, 2021

## 2.1 . Introduction

Gaussian process (GP) regression and interpolation (see, e.g., Rasmussen and Williams, 2006), also known as kriging (see, e.g., Stein, 1999), has gained significant popularity in statistics and machine learning as a non-parametric Bayesian approach for the prediction of unknown functions. The need for function prediction arises not only in supervised learning tasks, but also for building fast surrogates of time-consuming computations, e.g., in the assessment of the performance of a learning algorithm as a function of tuning parameters or, more generally, in the design and analysis computer experiments (Santner et al., 2003). The interest for GPs has also risen considerably due to the development of Bayesian optimization (Emmerich et al., 2006, Jones et al., 1998, Mockus, 1975, Srinivas et al., 2010...).

This context has fostered the development of a fairly large number of open-source packages to facilitate the use of GPs. Some of the popular choices are the Python modules scikit-learn (Pedregosa et al., 2011), GPy (Sheffield machine learning group, 2012–2020), GPflow (Matthews et al., 2017), GPyTorch (Gardner et al., 2018), OpenTURNS (Baudin et al., 2017); the R package DiceKriging (Roustant et al., 2012a); and the Matlab/GNU Octave toolboxes GPML (Rasmussen and Nickisch, 2010), STK (Bect et al., 2011–2021) and GPstuff (Vanhatalo et al., 2012).

In practice, all implementations require the user to specify the mean and covariance functions of a Gaussian process prior under a parameterized form. Out of the various methods available to estimate the model parameters, we can safely say that the most popular approach is the *maximum likelihood estimation* (MLE) method. However, a simple numerical experiment consisting in interpolating a function (see Table 2.1), as is usually done in Bayesian optimization, shows that different MLE implementations from different Python packages produce very dispersed numerical results when the default settings of each implementation are used. These significant differences were also noticed by Erickson et al. (2018) but the causes and possible mitigation were not investigated. Note that each package uses its own default algorithm for the optimization of the likelihood: GPyTorch uses ADAM (Kingma and Ba, 2015), OpenTURNS uses a truncated Newton method (Nash,

Table 2.1: Inconsistencies in the results across different Python packages. The results were obtained by fitting a GP model, with constant mean and a Matérn kernel ($\nu = 5/2$), to the Branin function, using the default settings for each package. We used $50$ training points and $500$ test points sampled from a uniform distribution on $[-5, 10] \times [0, 15]$. The table reports the estimated values for the variance and length scale parameters of the kernel, the empirical root mean squared prediction error (ERMSPE) and the minimized negative log likelihood (NLL). (Observe that the estimated length scale and variance parameters are very high, especially for GPy "improved". Such models are close to a spline (see, e.g., Barthelmé et al., 2022, Kimeldorf and Wahba, 1970).) The last row shows the improvement using the recommendations in this study.

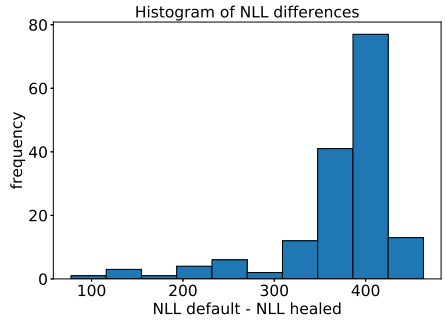
| LIBRARY | Version | Variance | Lengthscales | ERMSPE | NLL |
|---|---|---|---|---|---|
| SCIKIT-LEARN | 0.24.2 | $9.9 \cdot 10^4$ | $(13, 43)$ | 1.482 | 132.4 |
| GPY | 1.9.9 | $8.1 \cdot 10^8$ | $(88, 484)$ | 0.259 | 113.7 |
| GPYTORCH | 1.4.1 | $1.1 \cdot 10^1$ | $(4, 1)$ | 12.867 | 200839.7 |
| GPFLOW | 1.5.1 | $5.2 \cdot 10^8$ | $(80, 433)$ | 0.274 | 114.0 |
| OPENTURNS | 1.16 | $1.3 \cdot 10^4$ | $(8, 19)$ | 3.301 | 163.1 |
| GPY "IMPROVED" | 1.9.9 | $9.4 \cdot 10^{10}$ | $(220, 1500)$ | 0.175 | 112.0 |

1984) and the others generally use L-BFGS-B (Byrd et al., 1995). It turns out that none of the default results in Table 2.1 are really satisfactory compared to the result obtained using the recommendations in this study[1].

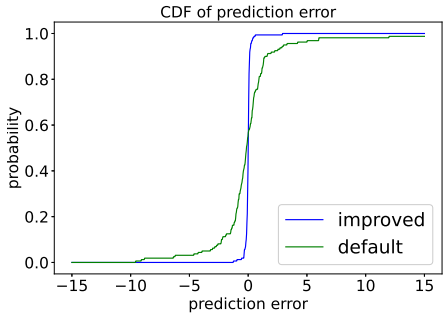
Focusing on the case of GP interpolation (with Bayesian optimization as the main motivation), the first contribution of this chapter is to understand the origin of the inconsistencies across available implementations. The second contribution is to investigate simple but effective strategies for improving these implementations, using the well-established GPy package as a case study. We shall propose recommendations concerning several optimization settings: initialization and restart strategies, parameterization of the covariance, etc. By anticipation of our numerical results, the reader is invited to refer to Figure 2.1 and Table 2.2, which show that significant improvement in terms of estimated parameter values and prediction errors can be obtained over default settings using better optimization schemes.

Even though this work targets a seemingly prosaic issue, and advocates somehow simple solutions, we feel that the contribution is nonetheless of significant value considering the widespread use of GP modeling. Indeed, a host of studies, particularly in the literature of Bayesian optimization, rely on off-the-shelf GP implementations: for their conclusions to be reliable and reproducible, robust im-

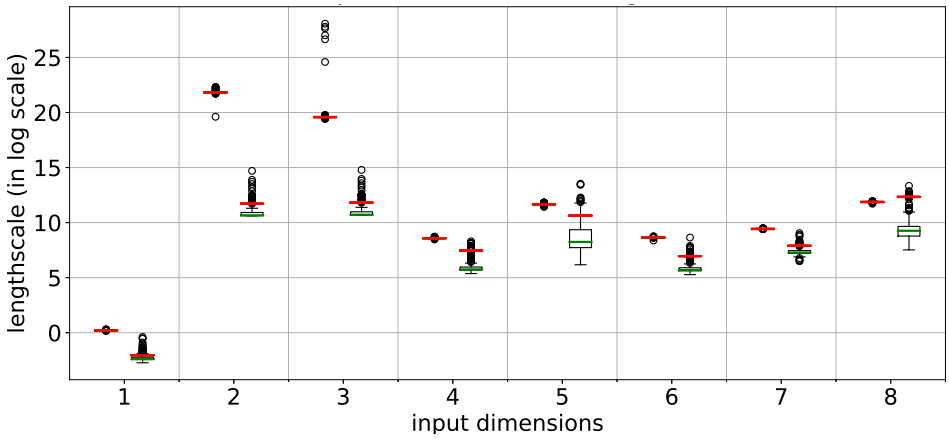
[1] Code available at https://github.com/saferGPMLE




(a) optimized NLL




(b) prediction error




(c) optimized lengthscales

Figure 2.1: *Improved* (cf. Section 2.6) vs *default* setups in GPy on the Borehole function with $n = 20d = 160$ random training points. We remove one point at a time to obtain (a) the distribution of the differences of negative log-likelihood (NLL) values between the two setups; (b) the empirical CDFs of the prediction error at the removed points; (c) pairs of box-plots for the estimated range parameters (for each dimension, indexed from 1 to 8 on the $x$-axis, the box-plot for *improved* setup is on the left and the box-plot for *default* setup is on the right; horizontal red lines correspond to the estimated values using the whole data set without leave-one-out). Notice that the parameter distributions of the *default* setup are more spread out.

Table 2.2: *Improved* (cf. Section 2.6) vs *default* setups in GPy for the interpolation of the Borehole function (input space dimension is $d = 8$) with $n \in \{3d, 5d\}$ random data points (see Section 2.5.3 for details). The experiment is repeated 50 times. The columns report the leave-one-out mean squared error (LOO-MSE) values (empirical mean over the repetitions, together with the standard deviation and the average proportion of the LOO-MSE to the total standard deviation of the data in parentheses).

| METHOD | $n = 3d$ | $n = 5d$ |
|---|---|---|
| DEFAULT | 17.559 (4.512, 0.387) | 10.749 (2.862, 0.229) |
| IMPROVED | 3.949 (1.447, 0.087) | 1.577 (0.611, 0.034) |

plementations are critical.

The chapter is organized as follows. Section 2.2 provides a brief review of GP modeling and MLE. Section 2.3 describes some numerical aspects of the evaluation and optimization of the likelihood function, with a focus on GPy's implementation. Section 2.4 provides an analysis of factors influencing the accuracy of numerical MLE procedures. Finally, Section 2.5 assesses the effectiveness of our solutions through numerical experiments and Section 2.6 concludes the chapter.

## 2.2 . Background

### 2.2.1 . Gaussian processes

Let $Z \sim \mathrm{GP}(m, k)$ be a Gaussian process indexed by $\mathbb{R}^d$, $d \geq 1$, specified by a mean function $m : \mathbb{R}^d \to \mathbb{R}$ and a covariance function $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$.

The objective is to predict $Z(x)$ at a given location $x \in \mathbb{R}^d$, given a data set $D = \{(x_i, z_i) \in \mathbb{R}^d \times \mathbb{R}, 1 \leq i \leq n\}$, where the observations $z_i$s are assumed to be the outcome of an additive-noise model: $Z_i = Z(x_i) + \varepsilon_i$, $1 \leq i \leq n$. In most applications, it is assumed that the $\varepsilon_i$s are zero-mean Gaussian i.i.d. random variables with variance $\sigma_\varepsilon^2 \geq 0$, independent of $Z$. (In rarer cases, heteroscedasticity is assumed.)

Knowing $m$ and $k$, recall (see, e.g. Rasmussen and Williams, 2006) that the posterior distribution of $Z$ is such that $Z \mid Z_1, \ldots, Z_n, m, k \sim GP(\widehat{Z}_n, k_n)$, where $\widehat{Z}_n$ and $k_n$ stand respectively for the posterior mean and covariance functions:

$$\begin{aligned} \widehat{Z}_n(x) &= m(x) + \textstyle\sum_{i=1}^n w_i(x;\underline{x}_n)\,(z_i - m(x_i))\,, \\ k_n(x,y) &= k(x,y) - \mathbf{w}(y;\underline{x}_n)^\mathsf{T}\mathbf{K}(\underline{x}_n, x)\,, \end{aligned}$$

where $\underline{x}_n$ denotes observation points $(x_1, \ldots, x_n)$ and the weights $w_i(x;\underline{x}_n)$ are so-

Table 2.3: Some kernel functions available in GPy. The Matérn kernel is recommended by Stein (1999). $\Gamma$ denotes the gamma function, $\mathscr{K}_\nu$ is the modified Bessel function of the second kind.

| KERNEL | $r(h)$, $h \in [0, +\infty)$ |
|---|---|
| SQUARED EXPONENTIAL | $\exp(-\frac{1}{2}r^2)$ |
| RATIONAL QUADRATIC | $(1+r^2)^{-\nu}$ |
| MATÉRN WITH PARAM. $\nu > 0$ | $\frac{2^{1-\nu}}{\Gamma(\nu)}\left(\sqrt{2\nu}r\right)^{\nu}\mathscr{K}_\nu\left(\sqrt{2\nu}r\right)$ |

lutions of the linear system:

$$(\mathbf{K}(\underline{x}_n, \underline{x}_n) + \sigma_\varepsilon^2 \mathbf{I}_n)\, \mathbf{w}(x; \underline{x}_n) = \mathbf{K}(\underline{x}_n, x)\,, \tag{2.1}$$

with $\mathbf{K}(\underline{x}_n, \underline{x}_n)$ the $n \times n$ covariance matrix with entries $k(x_i, x_j)$, $\mathbf{I}_n$ the identity matrix of size $n$, and $\mathbf{w}(x; \underline{x}_n)$ (resp. $\mathbf{K}(\underline{x}_n, x)$) the column vector with entries $w_i(x; \underline{x}_n)$ (resp. $k(x_i, x)$), $1 \leq i \leq n$.

It is common practice to assume a zero mean function $m = 0$—a reasonable choice if the user has taken care to center data—but most GP implementations also provide an option for setting a constant mean function $m(\cdot) = \mu \in \mathbb{R}$. In this chapter, we will include such a constant in our models, and treat it as an additional parameter to be estimated by MLE along with the others. (Alternatively, $\mu$ could be endowed with a Gaussian or improper-uniform prior, and then integrated out; see, e.g., O'Hagan (1978).)

The covariance function, aka covariance kernel, models similarity between data points and reflects the user's prior belief about the function to be learned. Most GP implementations provide a couple of stationary covariance functions taken from the literature (e.g., Rasmussen and Williams, 2006, Wendland, 2004). The *squared exponential*, the *rational quadratic* or the *Matérn* covariance functions are popular choices (see Table 2.3). These covariance functions include a number of parameters: a variance parameter $\sigma^2 > 0$ corresponding to the variance of $Z$, and a set of range (or length scale) parameters $\rho_1, \ldots, \rho_d$, such that

$$k(x, y) = \sigma^2 r(h)\,, \tag{2.2}$$

with $h^2 = \sum_{i=1}^{d} (x_{[i]} - y_{[i]})^2 / \rho_i^2$, where $x_{[i]}$ and $y_{[i]}$ denote the elements of $x$ and $y$. The function $r : \mathbb{R} \to \mathbb{R}$ in (2.2) is the stationary correlation function of $Z$. From now on, the vector of model parameters will be denoted by

$$\theta = (\sigma^2, \rho_1, \ldots, \rho_d, \ldots, \sigma_\varepsilon^2)^\mathsf{T} \in \Theta \subset \mathbb{R}^p,$$

and the corresponding covariance matrix $\mathbf{K}(\underline{x}_n, \underline{x}_n) + \sigma_\varepsilon^2 \mathbf{I}_n$ by $\mathbf{K}_\theta$.

### 2.2.2 . Maximum likelihood estimation

In this chapter, we focus on GP implementations where the parameters $(\theta, \mu) \in \Theta \times \mathbb{R}$ of the process $Z$ are estimated by maximizing the likelihood $\mathcal{L}(\underline{Z}_n|\theta,\mu)$ of $\underline{Z}_n = (Z_1, \ldots, Z_n)^\mathsf{T}$, or equivalently, by minimizing the negative log-likelihood (NLL)

$$-\log(\mathcal{L}(\underline{Z}_n|\theta,\mu)) = \frac{1}{2}(\underline{Z}_n - \mu\mathbb{1}_n)^\top \mathbf{K}_\theta^{-1}(\underline{Z}_n - \mu\mathbb{1}_n) + \frac{1}{2}\log|\mathbf{K}_\theta| + \text{constant}. \quad (2.3)$$

This optimization is typically performed by gradient-based methods, although local maxima can be of significant concern as the likelihood is often non-convex. Computing the likelihood and its gradient with respect to $(\theta, \mu)$ has a $O(n^3 + dn^2)$ computational cost (Petit et al., 2020a, Rasmussen and Williams, 2006).

## 2.3 . Numerical noise

The evaluation of the NLL as well as its gradient is subject to numerical noise, which can prevent proper convergence of the optimization algorithms. Figure 2.2 shows a typical situation where the gradient-based optimization algorithm stops before converging to an actual minimum. In this section, we provide an analysis on the numerical noise on the NLL using the concept of local condition numbers. We also show that the popular solution of adding *jitter* cannot be considered as a fully satisfactory answer to the problem of numerical noise.

Numerical noise stems from both terms of the NLL, namely $\frac{1}{2}\underline{Z}_n^\top \mathbf{K}_\theta^{-1}\underline{Z}_n$ and $\frac{1}{2}\log|\mathbf{K}_\theta|$. (For simplification, we assume $\mu = 0$ in this section.)

First, recall that the condition number $\kappa(\mathbf{K}_\theta)$ of $\mathbf{K}_\theta$, defined as the ratio $|\lambda_{\max}/\lambda_{\min}|$ of the largest eigenvalue to the smallest eigenvalue (Press et al., 1992), is the key element for analyzing the numerical noise on $\mathbf{K}_\theta^{-1}\underline{Z}_n$. In double-precision floating-point approximations of numbers, $\underline{Z}_n$ is corrupted by an error $\varepsilon$ whose magnitude is such that $\|\varepsilon\|/\|\underline{Z}_n\| \simeq 10^{-16}$. Worst-case alignment of $\underline{Z}_n$ and $\varepsilon$ with the eigenvectors of $\mathbf{K}_\theta$ gives

$$\frac{\|\mathbf{K}_\theta^{-1}\varepsilon\|}{\|\mathbf{K}_\theta^{-1}\underline{Z}_n\|} \simeq \kappa(\mathbf{K}_\theta) \times 10^{-16}, \quad (2.4)$$

which shows how the numerical noise is amplified when $\mathbf{K}_\theta$ becomes ill-conditioned.

The term $\log|\mathbf{K}_\theta|$ is nonlinear in $\mathbf{K}_\theta$, but observe, using $\mathrm{d}\log|\mathbf{K}_\theta|/\mathrm{d}\mathbf{K}_\theta = \mathbf{K}_\theta^{-1}$, that the differential of $\log|\cdot|$ at $\mathbf{K}_\theta$ is given by $H \mapsto \mathrm{Trace}(\mathbf{K}_\theta^{-1}H)$. Thus, the induced operator norm with respect to the Frobenius norm $\|\cdot\|_F$ is $\|\mathbf{K}_\theta^{-1}\|_F$. We can then apply results from Trefethen and Bau (1997) to get a local condition number of the mapping $\mathbf{A} \mapsto \log|\mathbf{A}|$ at $\mathbf{K}_\theta$:

$$\kappa(\log|\cdot|, \mathbf{K}_\theta) \triangleq \lim_{\varepsilon\to 0} \sup_{\|\delta_\mathbf{A}\|_F \le \varepsilon} \frac{\big|\log|\mathbf{K}_\theta + \delta_\mathbf{A}| - \log|\mathbf{K}_\theta|\big|}{\big|\log|\mathbf{K}_\theta|\big|} \frac{\|\mathbf{K}_\theta\|_F}{\|\delta_\mathbf{A}\|_F} = \frac{\sqrt{\sum_{i=1}^n \frac{1}{\lambda_i^2}}\sqrt{\sum_{i=1}^n \lambda_i^2}}{|\sum_{i=1}^n \log(\lambda_i)|} \quad (2.5)$$

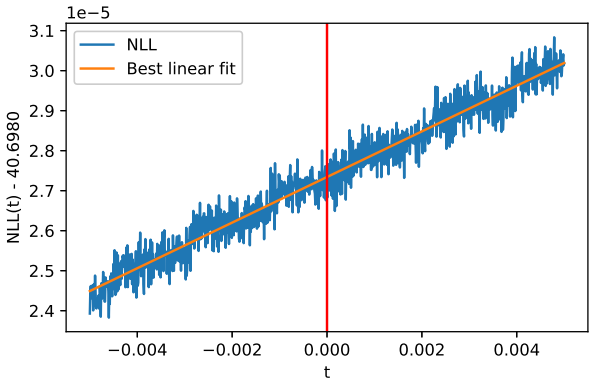



Figure 2.2: Noisy NLL profile along a particular direction in the parameter space, with a best linear fit (orange line). This example was obtained with GPy while estimating the parameters of a Matérn $5/2$ covariance, using $20$ data points sampled from a *Branin function*, and setting $\sigma_\varepsilon^2 = 0$. The red vertical line indicates the location where the optimization of the likelihood stalled.

where $\lambda_1, \cdots, \lambda_n$ are the (positive) eigenvalues of $\mathbf{K}_\theta$. Then, we have

$$\frac{\kappa(\mathbf{K}_\theta)}{|\sum_{i=1}^n \log(\lambda_i)|} \leq \kappa(\log|\cdot|, \mathbf{K}_\theta) \leq \frac{n\kappa(\mathbf{K}_\theta)}{|\sum_{i=1}^n \log(\lambda_i)|}, \tag{2.6}$$

which shows that numerical noise on $\log|\mathbf{K}_\theta|$ is linked to the condition number of $\mathbf{K}_\theta$.

The local condition number of the quadratic form $\frac{1}{2}\underline{Z}_n^\mathsf{T}\mathbf{K}_\theta^{-1}\underline{Z}_n$ as a function of $\underline{Z}_n$ can also be computed analytically. Some straightforward calculations show that it is bounded by $\kappa(\mathbf{K}_\theta)$.

(When the optimization algorithm stops in the example of Figure 2.2, we have $\kappa(\mathbf{K}_\theta) \simeq 10^{11}$ and $\kappa(\log|\cdot|, \mathbf{K}_\theta) \simeq 10^{9.5}$. The empirical numerical fluctuations are measured as the residuals of a local second-order polynomial best fit, giving noise levels $10^{-7}$, $10^{-8}$ and $10^{-7.5}$ for $\mathbf{K}_\theta^{-1}\underline{Z}_n$, $\frac{1}{2}\underline{Z}_n^\mathsf{T}\mathbf{K}_\theta^{-1}\underline{Z}_n$ and $\log|\mathbf{K}_\theta|$ respectively. These values are consistent with the above first-order analysis.)

Thus, when $\kappa(\mathbf{K}_\theta)$ becomes large in the course of the optimization procedure, numerical noise on the likelihood and its gradient may trigger an early stopping of the optimization algorithm (supposedly when the algorithm is unable to find a proper direction of improvement). It is well-known that $\kappa(\mathbf{K}_\theta)$ becomes large when $\sigma_\varepsilon^2 = 0$ and one of the following conditions occurs: 1) data points are close, 2) the covariance is very smooth (as for instance when considering the squared exponential covariance), 3) when the range parameters $\rho_i$ are large. These conditions arise more often than not. Therefore, the problem of numerical noise in the evaluation of the likelihood and its gradient is a problem that should not be neglected in GP implementations.

Table 2.4: Influence of the jitter on the GP model (same setting as in Figure 2.2). The table reports the condition numbers $\kappa(\mathbf{K}_\theta)$ and $\kappa(\log|\cdot|, \mathbf{K}_\theta)$, and the impact on the relative empirical standard deviations $\delta_{\text{quad}}$ and $\delta_{\text{logdet}}$ of the numerical noise on $\underline{Z}_n^T \mathbf{K}_\theta^{-1} \underline{Z}_n$ and $\log|\mathbf{K}_\theta|$ respectively (measured using second-order polynomial regressions). As $\sigma_\varepsilon$ increases, $\delta_{\text{quad}}$ and $\delta_{\text{logdet}}$ decrease but the interpolation error $\sqrt{\text{SSR}/\text{SST}} = \sqrt{\frac{1}{n}\sum_{j=1}^n (Z_j - \widehat{Z}_n(x_j))^2}/\text{std}(Z_1,\ldots,Z_n)$ and the NLL increase. Reducing numerical noise while keeping good interpolation properties requires careful attention in practice.

| $\sigma_\varepsilon^2$ / $\sigma^2$ | 0.0 | $10^{-8}$ | $10^{-6}$ | $10^{-4}$ | $10^{-2}$ |
|---|---|---|---|---|---|
| $\kappa(\mathbf{K}_\theta)$ | $10^{11}$ | $10^{9}$ | $10^{7.5}$ | $10^{5.5}$ | $10^{3.5}$ |
| $\kappa(\log\|\cdot\|, \mathbf{K}_\theta)$ | $10^{9.5}$ | $10^{8.5}$ | $10^{6.5}$ | $10^{4.5}$ | $10^{2.5}$ |
| $\delta_{\text{quad}}$ | $10^{-8}$ $(=10^{11-19})$ | $10^{-9.5}$ $(=10^{9-18.5})$ | $10^{-10.5}$ $(=10^{7.5-18})$ | $10^{-12}$ $(=10^{5.5-17.5})$ | $10^{-14}$ $(=10^{3.5-17.5})$ |
| $\delta_{\text{logdet}}$ | $10^{-7.5}$ $(=10^{9.5-17})$ | $10^{-9}$ $(=10^{8.5-17.5})$ | $10^{-11}$ $(=10^{6.5-17.5})$ | $10^{-13.5}$ $(=10^{4.5-18})$ | $10^{-15.5}$ $(=10^{2.5-18})$ |
| $-\log(\mathcal{L}(\underline{Z}_n\|\theta))$ | 40.69 | 45.13 | 62.32 | 88.81 | 124.76 |
| $\sqrt{\text{SSR}/\text{SST}}$ | $3.3\cdot10^{-10}$ | $1.2\cdot10^{-3}$ | 0.028 | 0.29 | 0.75 |

The most classical approach to deal with ill-conditioned covariance matrices is to add a small positive number on the diagonal of the covariance matrix, called *jitter*, which is equivalent to assuming a small observation noise with variance $\sigma_\varepsilon^2 > 0$. In GPy for instance, the strategy consists in always setting a minimal jitter of $10^{-8}$, which is automatically increased by an amount ranging from $10^{-6}\sigma^2$ to $10^{-1}\sigma^2$ whenever the Cholesky factorization of the covariance matrix fails (due to numerical non-positiveness). The smallest jitter making $\mathbf{K}_\theta$ numerically invertible is kept and an error is thrown if no jitter allows for successful factorization. However, note that large values for the jitter may yield smooth, non-interpolating approximations, with possible unintuitive and undesirable effects (see Andrianakis and Challenor, 2012), and causing possible convergence problems in Bayesian optimization.

Table 2.4 illustrates the behaviour of GP interpolation when $\sigma_\varepsilon^2$ is increased. It appears that finding a satisfying trade-off between good interpolation properties and low numerical noise level can be difficult. Table 2.4 also supports the connection in (2.4) and (2.6) between noise levels and $\kappa(\mathbf{K}_\theta)$. In view of the results of Figure 2.1 based on the default settings of GPy and Table 2.4, we believe that adaptive jitter cannot be considered as a do-it-all solution.

## 2.4 . Strategies for improving likelihood maximization

In this section we investigate simple but hopefully efficient levers / strategies to improve available implementations of MLE for GP interpolation, beyond the control of the numerical noise on the likelihood using jitter. We mainly focus on 1) initialization methods for the optimization procedure, 2) stopping criteria, 3) the effect of "restart" strategies and 4) the effect of the parameterization of the covariance.

### 2.4.1 . Initialization strategies

Most GP implementations use a gradient-based local optimization algorithm to maximize the likelihood that requires the specification of starting/initial values for the parameters. In the following, we consider different initialization strategies.

*Moment-based initialization.* A first strategy consists in setting the parameters using empirical moments of the data. More precisely, assuming a constant mean $m = \mu$, and a stationary covariance $k$ with variance $\sigma^2$ and range parameters $\rho_1, \ldots, \rho_d$, set

$$\mu_{\text{init}} = \text{mean}\,(Z_1, \ldots, Z_n), \tag{2.7}$$

$$\sigma^2_{\text{init}} = \text{var}\,(Z_1, \ldots, Z_n), \tag{2.8}$$

$$\rho_{k,\text{init}} = \text{std}\,(x_{1,[k]}, \ldots, x_{n,[k]}), \quad k = 1, \ldots, d, \tag{2.9}$$

where mean, var and std stand for the empirical mean, variance and standard deviation, and $x_{i,[k]}$ denotes the $k$th coordinate of $x_i \in \mathbb{R}^d$. The rationale behind (2.9) (following, e.g., Rasmussen and Williams, 2006) is that the range parameters can be thought of as the distance one has to move in the input space for the function value to change significantly and we assume, a priori, that this distance is linked to the dispersion of data points.

In GPy for instance, the default initialization consists in setting $\mu = 0$, $\sigma^2 = 1$ and $\rho_k = 1$ for all $k$. This is equivalent to the *moment-based* initialization scheme when the data (both inputs and outputs) are centered and standardized. The practice of standardizing the input domain into a unit length hypercube has been proposed (see, e.g., Snoek et al., 2012) to deal with numerical issues that arise due to large length scale values.

*Profiled initialization.* Assume the range parameters $\rho_1, \ldots, \rho_d$ (and more generally, all parameters different from $\sigma^2$, $\sigma^2_\varepsilon$ and $\mu$) are fixed, and set $\sigma^2_\varepsilon = \alpha\sigma^2$, with a prescribed multiplicative factor $\alpha \geq 0$. In this case, the NLL can be optimized analytically w.r.t. $\mu$ and $\sigma^2$. Optimal values turn out to be the generalized least squares solutions

$$\mu_{\text{GLS}} = (\mathbb{1}_n^\mathsf{T} \mathbf{K}_{\tilde{\theta}}^{-1} \mathbb{1}_n)^{-1} \mathbb{1}_n^\mathsf{T} \mathbf{K}_{\tilde{\theta}}^{-1} \underline{Z}_n\,, \tag{2.10}$$

$$\sigma^2_{\text{GLS}} = \frac{1}{n} (\underline{Z}_n - \mu_{\text{GLS}}\,\mathbb{1}_n)^\mathsf{T} \mathbf{K}_{\tilde{\theta}}^{-1} (\underline{Z}_n - \mu_{\text{GLS}}\,\mathbb{1}_n)\,, \tag{2.11}$$

where $\tilde{\theta} = (\sigma^2, \rho_1, \ldots, \rho_d, \ldots, \sigma^2_\varepsilon)^\mathsf{T} \in \Theta$, with $\sigma^2 = 1$ and $\sigma^2_\varepsilon = \alpha$. Under the *profiled* initialization scheme, $\rho_1, \ldots, \rho_d$ are set using (2.9), $\alpha$ is prescribed according to user's preference, and $\mu$ and $\sigma^2$ are initialized using (2.10) and (2.11).

*Grid-search initialization.* *Grid-search* initialization is a *profiled* initialization with the addition of a grid-search optimization for the range parameters.

Define a nominal range vector $\rho_0$ such that

$$\rho_{0,[k]} = \sqrt{d}\left(\max_{1\leq i\leq n} x_{i,[k]} - \min_{1\leq i\leq n} x_{i,[k]}\right), \quad 1 \leq k \leq d.$$

Then, define a one-dimensional grid of size $L$ (e.g., $L = 5$) by taking range vectors proportional to $\rho_0$: $\{\alpha_1\rho_0, \ldots, \alpha_L\rho_0\}$, where the $\alpha_i$s range, in logarithmic scale, from a "small" value (e.g., $\alpha_1 = 1/50$) to a "large" value (e.g., $\alpha_L = 2$). For each point of the grid, the likelihood is optimized with respect to $\mu$ and $\sigma^2$ using (2.10) and (2.11). The range vector with the best likelihood value is selected. (Note that this initialization procedure is the default initialization procedure in the Matlab/GNU Octave toolbox STK.)

### 2.4.2 . Stopping condition

Most GP implementations rely on well-tested gradient-based optimization algorithms. For instance, a popular choice in Python implementations is to use the limited-memory BFGS algorithm with box constraints (L-BFGS-B; see Byrd et al., 1995) of the SciPy ecosystem. (Other popular optimization algorithms include the ordinary BFGS, truncated Newton constrained, SQP, etc.; see, e.g., Nocedal and Wright (2006a).) The L-BFGS-B algorithm, which belongs to the class of quasi-Newton algorithms, uses limited-memory Hessian approximations and shows good performance on non-smooth functions (Curtis and Que, 2015).

Regardless of which optimization algorithm is chosen, the user usually has the possibility to tune the behavior of the optimizer, and in particular to set the stopping condition. Generally, the stopping condition is met when a maximum number of iterations is reached or when a norm on the steps and/or the gradient become smaller than a threshold.

By increasing the strictness of the stopping condition during the optimization of the likelihood, one would expect better parameter estimations, provided the numerical noise on the likelihood does not interfere too much.

### 2.4.3 . Restart and multi-start strategies

Due to numerical noise and possible non-convexity of the likelihood with respect to the parameters, gradient-based optimization algorithms may stall far from the global optimum. A common approach to circumvent the issue is to carry out several optimization runs with different initialization points. Two simple strategies can be compared.

*Restart.* In view of Figure 2.2, a first simple strategy is to restart the optimization algorithm to clear its memory (Hessian approximation, step sizes...), hopefully allowing it to escape a possibly problematic location using the last best parameters as initial values for the next optimization run. The optimization can be restarted a number of times, until a budget $N_{\text{opt}}$ of restarts is spent or the best value for the likelihood does not improve.

Table 2.5: Two popular reparameterization mappings $\tau$, as implemented, for example, in GPy and STK respectively. For *invsoftplus*, notice parameter $s > 0$, which is introduced when input standardization is considered (see Section 2.5).

| REPARAM. | $\tau : \mathbb{R}_+^\star \to \mathbb{R}$ | $\tau^{-1} : \mathbb{R} \to \mathbb{R}_+^\star$ |
|---|---|---|
| INVSOFTPLUS($s$) | $\log(\exp(\theta/s) - 1)$ | $s\log(\exp(\theta') + 1)$ |
| LOG | $\log(\theta)$ | $\exp(\theta')$ |

*Multi-start.* Given an initialization point $(\theta_{\text{init}}, \mu_{\text{init}}) \in \Theta \times \mathbb{R}$, a multi-start strategy consists in running $N_{\text{opt}} > 1$ optimizations with different initialization points corresponding to perturbations of the initial point $(\theta_{\text{init}}, \mu_{\text{init}})$. In practice, we suggest the following rule for building the perturbations: first, move the range parameters around $(\rho_{1,\text{init}}, \ldots, \rho_{d,\text{init}})^T$ (refer to Section 2.5 for an implementation); then, propagate the perturbations on $\mu$ and $\sigma^2$ using (2.10) and (2.11). The parameter with the best likelihood value over all optimization runs is selected.

### 2.4.4 . Parameterization of the covariance function

In practice, the parameters of the covariance functions are generally positive real numbers $(\sigma^2, \rho_1, \rho_2, \ldots)$ and are related to scaling effects that act "multiplicatively" on the predictive distributions. Most GP implementations introduce a reparameterization using a monotonic one-to-one mapping $\tau : \mathbb{R}_+^\star \to \mathbb{R}$, acting component-wise on the positive parameters of $\theta$, resulting in a mapping $\tau : \Theta \to \Theta'$. Thus, for carrying out MLE, the actual criterion $J$ that is optimized in most implementations may then be written as

$$J : \theta' \in \Theta' \mapsto -\log(\mathcal{L}(\underline{Z}_n | \tau^{-1}(\theta'), \mu)) \,. \tag{2.12}$$

Table 2.5 lists two popular reparameterization mappings $\tau$.

The effect of reparameterization is to "reshape" the likelihood. Typical likelihood profiles using the *log* and the so-called *invsoftplus* reparameterizations are shown on Figure 2.3. Notice that the NLL may be almost flat in some regions depending on the reparameterization. Changing the shape of the optimization criterion, combined with numerical noise, may or may not facilitate the convergence of the optimization.

## 2.5 . Numerical study

### 2.5.1 . Methodology

The main metric used in this numerical study is based on empirical cumulative distributions (ECDFs) of differences on NLL values.

More precisely, consider $N+1$ optimization schemes $S_0, S_1, \ldots, S_N$, where $S_0$ stands for a "brute-force" optimization scheme based on a very large number of

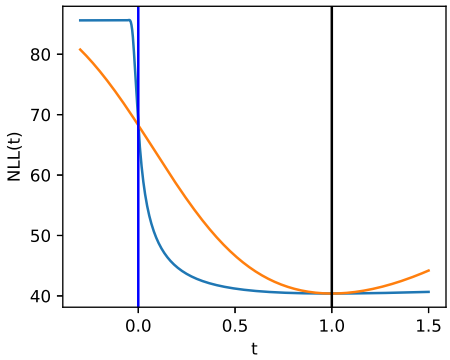



Figure 2.3: Profiles of the NLL along a linear path $t$ through the *profiled* initialization point (at zero, blue vertical line) and the optimum (at one, black vertical line). Orange (resp. blue) line corresponds to the *log* (resp. *invsoftplus*) reparameterization.

multi-starts, which is assumed to provide a robust MLE, and $S_1, \ldots, S_N$ are optimization schemes to be compared. Each optimization scheme is run on $M$ data sets $D_j$, $1 \leq j \leq M$, and we denote by $e_{i,j}$ the difference

$$e_{i,j} = \mathrm{NLL}_{i,j} - \mathrm{NLL}_{0,j}, \quad 1 \leq i \leq N, \quad 1 \leq j \leq M,$$

where $\mathrm{NLL}_{i,j}$ the NLL value obtained by optimization scheme $S_i$ on data set $D_j$.

A good scheme $S_i$ should concentrate the empirical distribution of the sample $E_i = \{e_{i,j}, j = 1, \ldots, M\}$ around zero—in other words, the ECDF is close to the ideal CDF $e \mapsto \mathbb{1}_{[0,\infty[}(e)$. Using ECDF also provides a convenient way to compare performances: a strategy with a "steeper" ECDF, or larger area under the ECDF, is better.

### 2.5.2 . Optimization schemes

All experiments are performed using GPy version 1.9.9, with the default L-BFGS-B algorithm. We use a common setup and vary the configurations of the optimization levers as detailed below.

*Common setup.* All experiments use an estimated constant mean-function, an anisotropic Matérn covariance function with regularity $\nu = 5/2$, and we assume no observation noise (the adaptive jitter of GPy ranging from $10^{-6}\sigma^2$ to $10^{2}\sigma^2$ is used, however).

*Initialization schemes.* Three initialization procedures from Section 2.4.1 are considered.

*Stopping criteria.* We consider two settings for the stopping condition of the L-BFGS-B algorithm, called *soft* (the default setting: `maxiter`$= 1000$, `factr`$= 10^7$, `pgtol`$= 10^{-5}$) and *strict* (`maxiter`$= 1000$, `factr`$= 10$, `pgtol`$= 10^{-20}$).

*Restart and multi-start.* The two strategies of Section 2.4.3 are implemented using a *log* reparameterization and initialization points $(\theta_{\text{init}}, \mu_{\text{init}})$ determined using a

*grid-search* strategy. For the *multi-start* strategy the initial range parameters are perturbed according to the rule $\rho \leftarrow \rho_{\text{init}} \cdot 10^{\eta}$ where $\eta$ is drawn from a $\mathcal{N}(0, \sigma_\eta^2)$ distribution. We take $\sigma_\eta = \log_{10}(5)/1.96$ $(\approx 0.35)$, to ensure that about 0.95 of the distribution of $\rho$ is in the interval $[1/5 \cdot \rho_{\text{init}},\ 5 \cdot \rho_{\text{init}}]$.

*Reparameterization.* We study the *log* reparameterization and two variants of the *invsoftplus*. The first version called *no-input-standardization* simply corresponds to taking $s = 1$ for each range parameter. The second version called *input-standardization* consists in scaling the inputs to a unit standard deviation on each dimension (by taking the corresponding value for $s$).

### 2.5.3 . Data sets

The data sets are generated from six well-known test cases in the literature of Bayesian optimization: the Branin function ($d = 2$; see, e.g. Surjanovic and Bingham, 2013), the Borehole function ($d = 8$; see, e.g. Worley, 1987), the Welded Beam Design problem ($d = 4$ and 6 outputs; see Chafekar et al., 2003), and the g10-ARH, the g10-RR, and the g10-PF problems (for a total of 13 unique outputs; see Section 1.6).

Each function is evaluated on Latin hypercube samples with a multi-dimensional uniformity criterion (LHS-MDU; Deutsch and Deutsch, 2012), with varying sample size $n \in \{3d, 5d, 10d, 20d\}$, resulting in a total of $21 \times 4 = 84$ data sets.

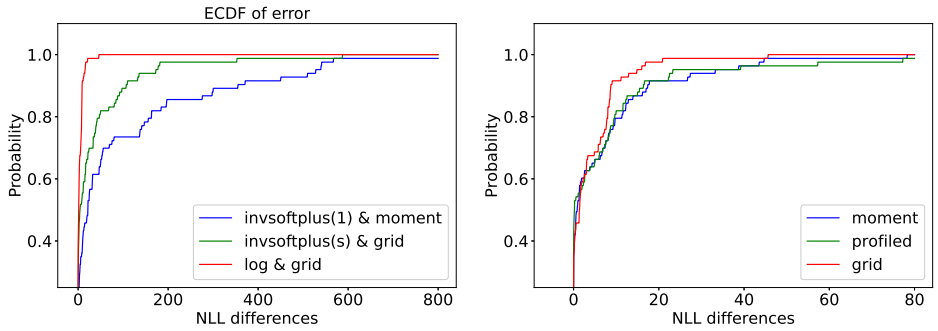
### 2.5.4 . Results and findings

Figure 2.4 shows the effect of reparameterization and the initialization method. Observe that the *log* reparameterization performs significantly better than the *invsoftplus* reparameterizations. For the *log* reparameterization, observe that the *grid-search* strategy brings a moderate but not negligible gain with respect to the two other initialization strategies, which behave similarly.

Next, we study the effect of the different restart strategies and the stopping conditions, on the case of the *log* reparameterization and *grid-search* initialization. The metric used for the comparison is the area under the ECDFs of the differences of NLLs, computed by integrating the ECDF between 0 and $\text{NLL}_{\text{max}} = 100$. Thus, a perfect optimization strategy would achieve an area under the ECDF equal to 100. Since the *multi-start* strategy is stochastic, results are averaged over 50 repetitions of the optimization procedures (for each $N_{\text{opt}}$ value, the optimization strategy is repeated 50 times). The areas are plotted against the computational run time. Run times are averaged over the repetitions in the case of the *multi-start* strategy.

Figure 2.5 shows that the *soft* stopping condition seems uniformly better. The *restart* strategy yields small improvements using moderate computational overhead. The *multi-start* strategy is able to achieve the best results at the price of higher computational costs.

## 2.6 . Conclusions and recommendations




(a) effect of reparameterization (b) effect of initialization

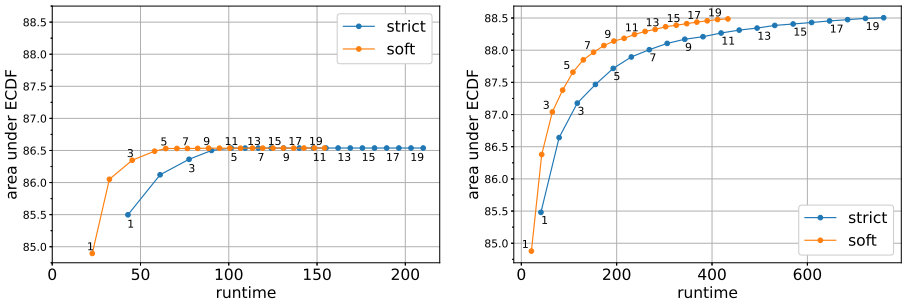
Figure 2.4: Initialization and reparameterization methods. (a) ECDFs corresponding to the best initialization method for each of the three reparameterizations—red line: *log* reparam. with *grid-search* init.; green line: *invsoftplus* with *input-standardization* reparam. and *grid-search* init; blue line: *invsoftplus* with *no-input-standardization* reparam. and *moment-based* init. (b) ECDFs for different initialization methods for the $\log$ reparameterization.

Our numerical study has shown that the parameterization of the covariance function has the most significant impact on the accuracy of MLE in GPy. Using *restart* / *multi-start* strategies is also very beneficial to mitigate the effect of the numerical noise on the likelihood. The two other levers have second-order but nonetheless measurable influence.

These observations make it possible to devise a recommended combination of improvement levers—for GPy at least, but hopefully transferable to other software packages as well. When computation time matters, an improved optimization procedure for MLE consists in choosing the combination of a *log* reparameterization, with a *grid-search* initialization, the *soft* (GPy's default) stopping condition, and a small number, say $N_{\text{opt}} = 5$, of restarts.

Figure 2.1 and Table 2.2 are based on the above optimization procedure, which results in significantly better likelihood values and smaller prediction errors. The *multi-start* strategy can be used when accurate results are sought.

As a conclusion, our recommendations are not intended to be universal, but will hopefully encourage researchers and users to develop and use more reliable and more robust GP implementations, in Bayesian optimization or elsewhere.




(a) restart with $N_{\text{opt}} = 1, \ldots, 20$

(b) multi-start with $N_{\text{opt}} = 1, \ldots, 20$, $\sigma_\eta = 0.35$

Figure 2.5: Area under the ECDF against run time: (a) *restart* strategy; (b) *multi-start* strategy. The maximum areas obtained are respectively 86.538 and 88.504.

# 3 - Efficient cross-validation for Gaussian process regression

This chapter is an article in preparation with Julien Bect and Emmanuel Vazquez extending (Petit et al., 2020a).

## 3.1 . Introduction

Gaussian process (GP) regression is a technique for inferring an unknown function using a limited number of possibly noisy observations (see, e.g., Rasmussen and Williams (2006), Santner et al. (2003), Stein (1999) and references therein). A GP specifies a prior distribution over functions by specifying a mean function and a covariance function. The standard practice is to select them from data within parametric families.

Maximum likelihood estimation is probably the most common technique for selecting the parameters. Cross-validation techniques are another common class of methods. In particular, cross-validation methods of the leave-one-out type are easy to implement due to the existence of fast formulas whose origin can be traced back to Allen (1971), and have been found again independently, extended or simply recalled in a number of works (see notably Bachoc, 2013a, Craven and Wahba, 1979, Dubrule, 1983, Ginsbourger and Schärer, 2021, Rasmussen and Williams, 2006, Sundararajan and Keerthi, 2001).

Fast formulas for leave-one-out prediction correspond to the following result. Let $Z = (Z_1, \ldots, Z_N)^\mathsf{T}$ be a zero-mean Gaussian vector with covariance matrix $K$. Then, $Z_i \mid Z_1, \ldots, Z_{i-1}, Z_{i+1}, \ldots, Z_n \sim \mathcal{N}\left(\widehat{Z}_i, \sigma_i^2\right)$, where $\widehat{Z}_i$ and $\sigma_i^2$ satisfy

$$\left\{ \begin{array}{rcl} Z_i - \widehat{Z}_i & = & \frac{\left(K^{-1}Z\right)_i}{\left(K^{-1}\right)_{i,i}}, \\ \sigma_i^2 & = & \frac{1}{\left(K^{-1}\right)_{i,i}}, \end{array} \right. \tag{3.1}$$

and where $(\cdot)_i$ and $(\cdot)_{i,j}$ denote vector and matrix entries. Identity (3.1) makes it possible to compute leave-one-out predictive distributions with considerable computational cost savings compared to a naive method that would recompute the predictive distribution for each observation removed from the standard kriging equations.

A technique for deriving $\widehat{Z}_i$ in (3.1) was first proposed by Craven and Wahba (1979, Lemma 3.2) and Dubrule (1983), and is based on the idea of replacing observations by predicted values. Only Dubrule (1983) provides the expression for $\sigma_i^2$. Note also that Dubrule (1983) extends (3.1) to the leave-$p$-out case. Another approach to derive (3.1), which is proposed by Rasmussen and Williams (2006, Section 5.4.2) for the leave-one-out case, and in Ginsbourger and Schärer (2021)

in the general case, is to resort to the blockwise inversion formula of a matrix $M$:

$$M^{-1} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}^{-1} = \begin{pmatrix} A^{-1} + A^{-1}B(M/A)^{-1}CA^{-1} & -A^{-1}B(M/A)^{-1} \\ -(M/A)^{-1}CA^{-1} & (M/A)^{-1} \end{pmatrix} \quad (3.2)$$

with $M/A = D - CA^{-1}B$ standing for the Schur complement of $A$ in $M$.

In this work, we propose another technique to derive (3.1), which is based on the well-known conditional distribution formulas for Gaussian random vectors, using the precision matrix $\Delta = K^{-1}$ (see, e.g., Von Mises, 1964, p. 200). We believe that our technique is simpler than the ones previously proposed, and provides moreover a better understanding of (3.1).

Fast formulas to obtain leave-one-out predictive distributions such as (3.1) make it possible to build goodness-of-fit criteria to select the parameters of a GP from data. This is usually done by finding the parameters of the covariance that maximize goodness of fit. We thus get to an optimization problem, for which having the gradients of the selection criterion at low computational cost is paramount. For leave-one-out criteria, Petit et al. (2020a) give efficient formulas for the gradients of (3.1), so that the computational cost of optimizing leave-one-out criteria is the same as that of maximum likelihood estimation.

In this work, we extend the formulas of Petit et al. (2020a) for obtaining the gradients of general cross-validation criteria, and make general complexity statements for $K$-fold and leave-$p$-out cross-validation schemes.

This chapter is organized as follows. Section 3.2 is about obtaining fast formulas for cross-validation using precision matrices. Section 3.3 gives the gradients of cross-validation formulas at low computational cost. Finally, Section 3.4 illustrates the interest of non-standard cross-validation schemes with an example of $K$-fold cross-validation.

## 3.2 . Fast formulas for cross-validation

In this section, we consider more generally a random vector $Z \in \mathbb{R}^N$ such that $Z \mid \beta \sim \mathcal{N}(\Phi\beta, K)$, where $\Phi$ is a known matrix of size $N \times q$, $\beta \sim \mathcal{N}(b, B)$ is a Gaussian random vector of size $q$, and $K$ is a known covariance matrix assumed invertible.

The matrix $\Delta = K^{-1}$ is called the precision matrix of $Z$. The use of precision matrices is ubiquitous in the field of Gaussian Markov random fields (see, e.g., Rue and Held, 2005, Section 2.2), since precision matrices provide a simple way to state conditional independence properties. The derivation (3.4)–(3.9) is available in numerous textbooks and articles, but we repeat it for completeness. However, its link to the fast cross-validation formulas (3.1) is new, to the best of our knowledge.

Assume first that $Z \sim \mathcal{N}(0, K)$, which corresponds to the limit case $B \to 0$, and consider a decomposition $Z = \left(Z_0^\mathsf{T}, Z_1^\mathsf{T}\right)^\mathsf{T}$, with $Z_0 \in \mathbb{R}^n$ and $Z_1 \in \mathbb{R}^{N-n}$, $1 \leq n \leq N$.

Then, using the corresponding block decomposition

$$\Delta = \begin{pmatrix} \Delta_{0,0} & \Delta_{0,1} \\ \Delta_{1,0} & \Delta_{1,1} \end{pmatrix}, \tag{3.3}$$

the conditional density of $Z_1$ given $Z_0$ can be written as

$$\begin{aligned} p(Z_1 \,|\, Z_0) = \; & p(Z_0, Z_1)/p(Z_0) && (3.4)\\ \propto \; & \exp\Big(-\frac{1}{2}(Z_0^\mathsf{T}, Z_1^\mathsf{T}) \begin{pmatrix} \Delta_{0,0} & \Delta_{0,1} \\ \Delta_{1,0} & \Delta_{1,1} \end{pmatrix} (Z_0^\mathsf{T}, Z_1^\mathsf{T})^\mathsf{T}\Big) && (3.5)\\ \propto \; & \exp\Big(-\frac{1}{2} Z_1^\mathsf{T} \Delta_{1,1} Z_1 - Z_0^\mathsf{T} \Delta_{0,1} \Delta_{1,1}^{-1} \Delta_{1,1} Z_1\Big) && (3.6)\\ = \; & \exp\Big(-\frac{1}{2} Z_1^\mathsf{T} \Delta_{1,1} Z_1 + \widehat{Z}_1^\mathsf{T} \Delta_{1,1} Z_1\Big) && (3.7)\\ \propto \; & \exp\Big(-\frac{1}{2} (Z_1 - \widehat{Z}_1)^\mathsf{T} \Delta_{1,1} (Z_1 - \widehat{Z}_1)\Big), && (3.8) \end{aligned}$$

with $\widehat{Z}_1 = -\Delta_{1,1}^{-1}\Delta_{1,0} Z_0$. Thus, we have

$$Z_1 \,|\, Z_0 \sim \mathcal{N}(\widehat{Z}_1, \Delta_{1,1}^{-1}). \tag{3.9}$$

Then, a generalization of the fast cross-validation formulas (3.1) can be written as

$$Z_1 - \widehat{Z}_1 = \Delta_{1,1}^{-1}\Delta_{1,1} Z_1 + \Delta_{1,1}^{-1}\Delta_{1,0} Z_0 = \Delta_{1,1}^{-1} [\Delta Z]_1 \,, \tag{3.10}$$

where $[\Delta Z]_1$ denotes the subvector corresponding to last $N-n$ entries of $\Delta Z$.

The following proposition extends (3.9) and (3.10) when $B \neq 0$. It is an extension of (Bachoc, 2013a, Proposition 2.35) to more general cross-validation schemes than leave-one-out. In addition, our proof is new, to the best of our knowledge.

**Proposition 11** *Let $Z \in \mathbb{R}^N$ be a random vector such that $Z \,|\, \beta \sim \mathcal{N}(\Phi\beta, K)$, where $\Phi$ is a known matrix of size $N \times q$ with rank $q$, $\beta \sim \mathcal{N}(b, B)$ is a Gaussian random vector of size $q$, and $K$ is a known covariance matrix of size $N \times N$ assumed invertible.*

*Consider a decomposition $Z = \left(Z_0^\mathsf{T}, Z_1^\mathsf{T}\right)^\mathsf{T}$, with $Z_0 \in \mathbb{R}^n$ and $Z_1 \in \mathbb{R}^{N-n}$, $1 \leq n \leq N$. Then,*

$$p(Z_1 \,|\, Z_0) = \mathcal{N}\left(\widehat{Z}_1, \Delta_{1,1}^{-1}\right), \tag{3.11}$$

*with*

$$\widehat{Z}_1 = \mu_1 - \Delta_{1,1}^{-1}\Delta_{1,0}(Z_0 - \mu_0),$$

*where $\mu_0$ (respectively $\mu_1$) corresponds to the $n$ first entries (respectively $n - N$ last entries) of $\mu \in \mathbb{R}^N$ and $\Delta_{0,0}, \Delta_{0,1}, \Delta_{1,0}$ and $\Delta_{1,1}$ correspond to the block decomposition as in (3.3) of $\Delta$, and where the vector $\mu$ and the matrix $\Delta$ are defined according to three cases as follows.*

*Case 1: If $\beta = b$ is a deterministic parameter ($B = 0$), then*

$$\begin{cases} \mu = \Phi b, \\ \Delta = K^{-1}. \end{cases}$$

*Case 2: If $\beta \sim \mathcal{N}(b, B)$, then*

$$\begin{cases} \mu = \Phi b, \\ \Delta = \left(K + \Phi B \Phi^{\mathsf{T}}\right)^{-1}. \end{cases}$$

*Case 3: Using the improper prior density $p(\beta) \propto 1$ (which can be seen as a limit of Case 2, when $B = s^2 I_q$ and $s^2 \to \infty$), we have*

$$\begin{cases} \mu = 0, \\ \Delta = K^{-1} - K^{-1}\Phi\left(\Phi^{\mathsf{T}} K^{-1} \Phi\right)^{-1} \Phi^{\mathsf{T}} K^{-1}, \end{cases}$$

*assuming a proper posterior distribution.*

**Proof**

Case 1 is obtained from Case 2 with $B = 0$.

In Case 2, $Z - \Phi b$ has zero mean and a covariance matrix equal to $K + \Phi B \Phi^{\mathsf{T}}$. Thus, (3.9) applies.

In Case 3, the posterior density of $Z_1$ and $\beta$ is

$$p(Z_1, \beta \,|\, Z_0) = \frac{p(Z, \beta)}{p(Z_0)} \propto p(Z, \beta) \propto p(Z \,|\, \beta) = \exp\left\{-(Z - \Phi\beta)^{\mathsf{T}} K^{-1} (Z - \Phi\beta)\right\}$$

By integration, using the identity for multivariate normal pdfs in Lemma B.1 of Santner et al. (2003), we obtain

$$\begin{aligned} p(Z_1 \,|\, Z_0) = & \int p(Z_1, \beta \,|\, Z_0)\, \mathrm{d}\beta \\ & \propto \exp\left\{-\frac{1}{2} Z^{\mathsf{T}} \left(K^{-1} - K^{-1}\Phi\left(\Phi^{\mathsf{T}} K^{-1}\Phi\right)^{-1} \Phi^{\mathsf{T}} K^{-1}\right) Z\right\}. \end{aligned}$$

Then, the calculation (3.5)–(3.8) can be repeated with

$$\Delta = K^{-1} - K^{-1}\Phi\left(\Phi^{\mathsf{T}} K^{-1}\Phi\right)^{-1} \Phi^{\mathsf{T}} K^{-1}.$$

■

Table 3.1: Complexities of computing $\widehat{Z}_i$ and $\Sigma_i$ for different cross-validation schemes (see, e.g., Arlot and Celisse, 2009). The leave-$p$-out cross-validation scheme stands for taking $I = \binom{N}{p}$ and all the subsets of size $p$ for $T_1, \ldots, T_I$.

| | Leave-$p$-out | $K$-fold | General case |
|---|---|---|---|
| Naive | $\mathcal{O}\left(n^{p+3}\right)$ | $\mathcal{O}\left(Kn^3\right)$ | $\mathcal{O}\left(\sum_{i=1}^{I}\sum_{q=0}^{3}\lvert T_i\rvert^q(n-\lvert T_i\rvert)^{3-q}\right)$ |
| Fast formulas | $\mathcal{O}\left(n^{\max(3,p+1)}\right)$ | $\mathcal{O}\left(n^3\right)$ | $\mathcal{O}\left(n^3+\sum_{i=1}^{I}\lvert T_i\rvert^3+\lvert T_i\rvert^2 n\right)$ |

## 3.3 . Efficient cross-validation schemes for model selection in GP regression

### 3.3.1 . Complexity of cross-validation schemes

Proposition 11 gives the predictive distribution for a *hold-out* cross-validation scheme, which forms the building block of more complicated schemes (see, e.g., Arlot and Celisse, 2009).

Let $T_1, \cdots, T_I$ be non-trivial subsets of $\{1, \cdots, N\}$. Denote by $Z_1 \in \mathbb{R}^{|T_1|}$, $\ldots$, $Z_I \in \mathbb{R}^{|T_I|}$ the associated sub-vectors of $Z \in \mathbb{R}^N$ and denote by $Z_{-1} \in \mathbb{R}^{N-|T_1|}$, $\ldots$, $Z_{-I} \in \mathbb{R}^{N-|T_I|}$ the sub-vectors corresponding to the complements of the sets $T_1, \cdots, T_I$. For instance, in the case of a leave-one-out cross-validation scheme, we have $I = N$ and $T_i = \{i\}$, $i = 1, \ldots, N$.

Our objective is to construct a Gaussian model for $Z$, such that for each $i$, $\mathcal{N}(\widehat{Z}_i, \Sigma_i)$, with $\widehat{Z}_i$ and $\Sigma_i$ obtained from (3.11) (with $Z_0 \leftarrow Z_{-i}$ and $Z_1 \leftarrow Z_i$), is a good predictive distribution for $Z_i$.

Petit et al. (2021a) consider goodness-of-fit criteria written as:

$$\sum_{i=1}^{I} S_i\left(\mathcal{N}\left(\widehat{Z}_i, \Sigma_i\right), Z_i\right), \tag{3.12}$$

where and $S_i \colon \mathscr{P}_i \times \mathbb{R}^{|T_i|}$, $i = 1, \ldots, I$, are (negatively-oriented) *scoring rules* (see, e.g., Gneiting and Raftery, 2007), i.e., mappings such that $S_i(P, z)$ expresses a loss for predicting $z \in \mathbb{R}^{|T_i|}$ using $P$ in a class $\mathscr{P}_i$ of predictive distributions. Ubiquitous examples of scoring rules for GP regression are the squared prediction error $S_{\mathrm{SPE}}(P, z) = \|z - \gamma\|^2$, with $\gamma$ the mean of $P$, and the negative log predictive density $S_{\mathrm{NLPD}}(P, z) = -\log(p(z))$, where $p$ is the probability density of $P$.

The complexities for computing the predictive moments $\widehat{Z}_i$ and $\Sigma_i$ using the standard kriging equations for each hold-out or using the fast formulas (3.11) are given in Table 3.1. A gain up to two orders of magnitude for the leave-$p$-out and can be achieved. For $K$-fold cross-validation, the gain is a factor $K$.

### 3.3.2 . Efficient computation of gradients

Suppose that the covariance matrix $K$ of $Z$ is parametrized by $\omega \in \Omega \subseteq \mathbb{R}^p$. Define $\theta = \left(\omega^{\mathsf{T}}, b^{\mathsf{T}}\right)^{\mathsf{T}}$, if $\beta$ is treated as a deterministic parameter $b$, or $\theta = \omega$

otherwise. Using the notations of Section 3.3.1, the selection of $\theta$ can be carried out by optimizing the criterion

$$L(\theta) = \sum_{i=1}^{I} S_i\left(\mathscr{N}\left(\widehat{Z}_i, \Sigma_i\right), Z_i\right), \tag{3.13}$$

where $\widehat{Z}_i$ and $\Sigma_i$ depend on $\theta$.

In practice, the criterion (3.13) is optimized using a gradient-based algorithm. In the following, we extend the approach by Petit et al. (2020a) and show how to compute the gradient of $L$ using *reverse mode differentiation* (Linnainmaa, 1970), also known as *back-propagation*. First, notice that $L$ can be written as the composition of three operators: $L = \varphi \circ \rho \circ \varsigma$ with

$$\begin{cases} \varsigma\colon \theta \in \mathbb{R}^q \mapsto (\mu, K) \in \mathbb{R}^N \times S_{++}^N, \\ \rho\colon (\mu, K) \in \mathbb{R}^N \times S_{++}^N \mapsto \Gamma = \left(\widehat{Z}_i, \Sigma_i\right)_i \in \prod_{i=1}^I \mathbb{R}^{|T_i|+|T_i|^2}, \\ \varphi\colon \Gamma \in \prod_{i=1}^I \mathbb{R}^{|T_i|+|T_i|^2} \mapsto L \in \mathbb{R}, \end{cases} \tag{3.14}$$

where $S_{++}^N$ is the space of strictly positive definite symmetric matrices and $\mu$ is defined in Proposition 11, according to the assumption on $\beta$.

The gradient of $L$ can be computed using the chain rule

$$J_L = J_\varphi J_\rho J_\varsigma, \tag{3.15}$$

where the symbol $J$ stands for a Jacobian matrix. In the particular case of leave-one-out cross-validation, Petit et al. (2020a) argue that the computational cost of (3.15) depends strongly on its implementation.

Computing $J_\varsigma$ is discussed by Petit et al. (2020a), and the case of $J_\varphi$ is postponed to the end of this section. In the case of leave-one-out cross-validation with $\mu = 0$, Petit et al. (2020a) use reverse-mode differentiation to get an efficient computation of the adjoint

$$\mathscr{L}_\rho^*\colon \delta^\Gamma \mapsto J_\rho^\mathsf{T} \delta^\Gamma \tag{3.16}$$

of $\rho$. $\mathscr{L}_\rho^*$ expresses how a variation $\delta^\Gamma$ on the value of $\Gamma$ of is (back-)propagated on $\mu$ and $K$. It can be used to compute $\mathscr{L}_\rho^*\left(J_\varphi^\mathsf{T}\right) = \left(J_\varphi J_\rho\right)^\mathsf{T}$ in (3.15). More precisely, Algorithm 1 is an efficient implementation of $\mathscr{L}_\rho^*$, which can be used for any cross-validation scheme 3.13. By counting elementary operations, observe that the computation of the adjoint of $\rho$ using Algorithm 1 has the same order of algorithmic complexity than the computation of $\rho$ itself (as shown in Table 3.1). Moreover, the complexity of the computation of $\rho$ and its adjoint dominates that of other operations.

**Algorithm 1** Computing $\delta^\mu$ and $\delta^K$ from $\delta^\Gamma = (\delta^{\widehat{Z}_i}, \delta^{\Sigma_i},\ 1 \leq i \leq I)$. Given a vector $a$ (resp. a matrix $A$) and $1 \leq i, j \leq I$, $a_i$ (resp. $A_{i,j}$) stands for the subvector (resp. the submatrix) with indices in $T_i$ (resp. in $T_i \times T_j$). Furthermore, the subscript ":" along a dimension indicates taking all the indices.

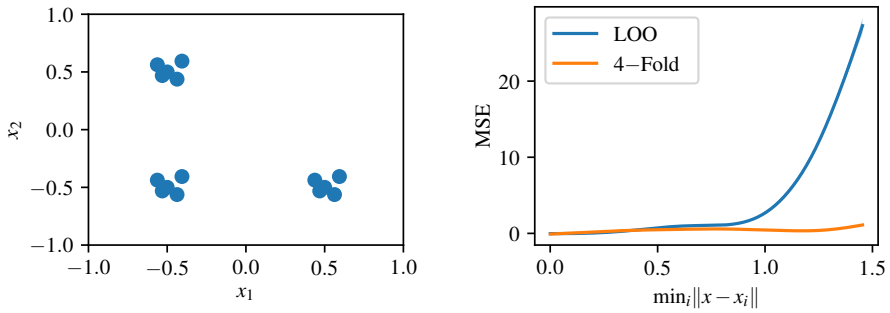
**Input:** $\mu$, $K$, $\delta^\Gamma = (\delta^{\widehat{Z}_i}, \delta^{\Sigma_i},\ 1 \leq i \leq I)$,
Compute $\Delta$ from $K$ and $\Phi$ using Proposition 11
$\alpha = \Delta Z$
$\delta^\mu = 0$, $\delta^K = 0$, $\delta^\Delta = 0$
**for** $i = 1$ **to** $I$ **do**
  $\delta^{\Delta_{i,i}^{-1}} = \delta^{\Sigma_i} - \delta^{\widehat{Z}_i} \alpha_i^\mathsf{T}$
  $\delta^{\alpha_i} = -\Delta_{i,i}^{-1} \delta^{\widehat{Z}_i}$
  $\delta^\mu = \delta^\mu - \Delta_{:,i} \delta^{\alpha_i}$
  $\delta^\Delta_{i,:} = \delta^\Delta_{i,:} + \delta^{\alpha_i} (Z - \mu)^\mathsf{T}$
  $\delta^\Delta_{i,i} = \delta^\Delta_{i,i} - \Delta_{i,i}^{-1} \delta^{\Delta_{i,i}^{-1}} \Delta_{i,i}^{-1}$
**end for**
$\delta^K = -\Delta \delta^\Delta \Delta$

**Algorithmic complexity of the computation of $\varphi$** Regarding the computation of (3.12) from the $\widehat{Z}_{I_i}$s and $\Sigma_{I_i}$, assume that we have a sequence $(S_m)_{m \geq 1}$ of scoring rules such that $S_m \colon \mathscr{P}_m \times \mathbb{R}^m$ can be computed along with its gradient in $g(m)$ time. Then using leave-$p$-out yields $\mathscr{O}(g(p) n^p)$ time, which can always be neglected compared to the complexity of $\rho$ given by Table 3.1, whereas using $K$-fold yields $\mathscr{O}(g(n/K))$ time, which does not change the overall complexity when $g(m) = \mathscr{O}(m^3)$. (This is the case of the scoring rules $S_{\mathrm{SPE}}$ and $S_{\mathrm{NLPD}}$ mentioned in Section 3.3.1.)

**Comparison to maximum likelihood estimation** Petit et al. (2020a) pointed out that $\mathscr{O}(dn^2)$ time implementations already exist for $\varsigma$ and its Jacobian matrix in the typical case of a homoscedastic noise, a constant mean, and a family of covariance functions with one parameter per dimension and a fixed number of remaining parameters. In this case, they conclude that the likelihood criterion and any leave-one-out criterion can be evaluated along with its gradients in $\mathscr{O}\left(n^3 + dn^2\right)$ time.

Our contribution shows that the same holds for the leave-2-out and $K$-fold criteria—provided that $g(m) = \mathscr{O}(m^3)$ for the latter. However, for leave-$p$-out with $p \geq 3$, the cost is increased to $\mathscr{O}\left(n^{p+1} + dn^2\right)$.

## 3.4 . Numerical experiments




(a) Design (b) Error

Figure 3.1: (a) a design of experiments for the function $f$, composed of three clusters of five points. (b) a local polynomial regression of the SPE versus the distance of the location $x$ to be predicted to the design of experiments.

In the spirit of Ginsbourger and Schärer (2021), this section illustrates the interest of using other cross-validation schemes than leave-one-out for GP regression. We consider the test function

$$f : x \in [-1,1]^2 \mapsto \sin(4\|x\|)$$

evaluated on a non-uniform design of size $n = 15$, as shown in Figure 3.1(a). Notice that there are three clusters of points. The prediction of $f$ is carried out using a noiseless zero-mean GP model $\xi \sim \mathrm{GP}(0,k)$, with an isotropic Matérn ($\nu = 5/2$) covariance function $k$ of the form $\mathrm{cov}(\xi(x),\xi(y)) = \sigma^2 r(\frac{\|x-y\|}{\rho})$ (see, e.g., Matérn, 1986, Stein, 1999).

We study the selection of the length scale parameter $\rho > 0$ using two selection criteria based on $S_{\mathrm{SPE}}$: the standard leave-one-out criterion and a 4-fold criterion, with one fold per cluster of design points. The two criteria have the same order of complexity, according to Table 3.1, but lead to different length scale estimates on this example.

The leave-one-out criterion yields a large value for $\rho$ ($\rho = 91.1$), whereas the 4-fold criterion leads to a much smaller estimate ($\rho = 0.179$).

If we use the selected models for making predictions on a space-filling design of size $10^4$, the leave-one-out criterion yields a mean squared error of $0.71$ and the 4-fold criterion yields $0.37$. Furthermore, a more careful analysis of the error is provided in Figure 3.1(b). It reveals that the 4-fold criterion yields a model that is more precise for making predictions in empty regions.

# Part III

# Goal-oriented modeling

# 4 - Relaxed Gaussian process interpolation: a goal-oriented approach to Bayesian optimization

This chapter is a reproduction of Petit et al. (2022b) with few modifications.

S. J. Petit, J. Bect, and E. Vazquez. Relaxed Gaussian process interpolation: a goal-oriented approach to Bayesian optimization. 2022b. URL https://arxiv.org/abs/2107.06006

## 4.1 . Introduction

### 4.1.1 . Context and motivation

Gaussian process (GP) interpolation and regression (see, e.g., Rasmussen and Williams, 2006, Stein, 1999) is a very classical method for predicting an unknown function from data. It has found applications in active learning techniques, and notably in Bayesian optimization, a popular derivative-free global optimization technique for functions whose evaluations are time-consuming.

A GP model is defined by a mean and a covariance functions, which are generally selected from data within parametric families. The most popular models assume stationarity and rely on standard covariance functions such as the Matérn covariance. The assumption of stationarity yields models with relatively low-dimensional parameters. However, such a hypothesis can sometimes result in poor models when the function to be predicted has different scales of variation or different local regularities across the domain.

This is the case for instance in the motivating example given by Gramacy and Lee (2008), or in the even simpler toy minimization problem shown in Figure 4.1. The objective function in this example, which we shall call the Steep function, is smooth with an obvious global minimum around the point $x = 8$. However, the variations around the minimum are overshadowed by some steep variations on the left. Figure 4.2 shows a stationary GP fit with $n = 8$ points, where the parameters of the covariance function have been selected using maximum likelihood. Observe that the confidence bands are too large and that the conditional mean varies too much in the neighborhood of the global minimum, consistently with the stationary GP model that reflects the prior that our function oscillates around a mean value with a constant scale of variations. In this case, even if GP interpolation is consistent (Vazquez and Bect, 2010a), stationarity seems an unsatisfactory assumption for the Steep function. One expects Bayesian optimization techniques to be somehow inefficient on this problem with such a stationary model, whose posterior distributions are too pessimistic in the region of the minimum.

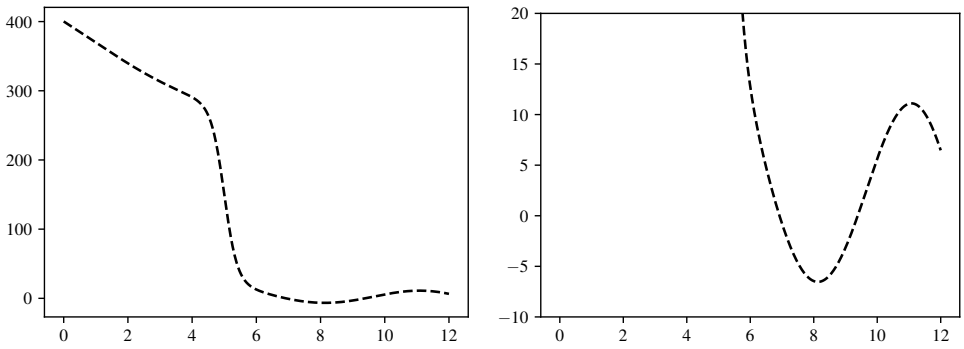



Figure 4.1: Left: the Steep function. Right: same illustration with a restrained range on the $y$-axis. The variations on the left overshadow the global minimum on the right.

Nevertheless, the Steep function has the characteristics of an easy optimization problem: it has only two local minima, with the global minimum lying in a valley of significant volume. Consequently, a Bayesian optimization technique could be competitive if it relied on a model giving good predictions in regions where the function takes low values. In this work, we propose to explore goal-oriented GP modeling, where we want predictive models in regions of interest, even if it means being less predictive elsewhere.

### 4.1.2 . Related works

Going beyond the stationary hypothesis has been an active direction of research. With maybe a little bit of oversimplification, one can distinguish two categories of approaches that all use stationary Gaussian processes as a core building block: local models and transformation/composition of models.

#### Local models

A first class of local models is obtained by considering partitions of the input domain with different GP models on each subset. Partitions can be built by splitting the domain along the coordinate axes. This is the case of the treed Gaussian process models proposed by Gramacy and Lee (2008), which combines a fully Bayesian framework and the use of RJ-MCMC techniques for the inference, or the trust-region method by Eriksson et al. (2019). Park and Apley (2018) also propose partition-based local models built by splitting the domain along principal component directions. In such techniques, there are parameters (often many of them) related to, e.g., the way the partitions evolve with the data, the size of the partitions, or how local Gaussian processes interact with each other.

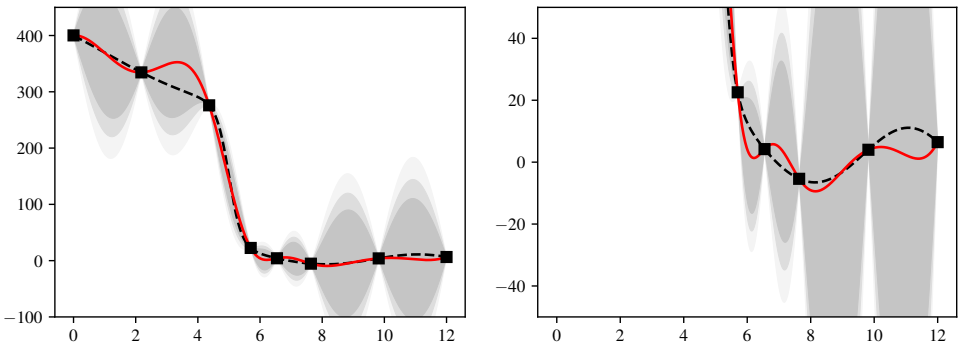



Figure 4.2: Left: GP fit on the Steep function. Right: same illustration with a restrained range on the $y$-axis. The squares represent the data. The red line represents the posterior mean $\mu_n$ given by the model and the gray envelopes represent the associated uncertainties.

A second class of local models is obtained by spatially weighting one or several GP models. Many schemes have been proposed, including methods based on partition of unity (Nott and Dunsmuir, 2002), weightings of covariance functions (Pronzato and Rendas, 2017, Rivoirard and Romary, 2011), and convolution techniques (see, e.g., Gibbs, 1998, Higdon, 1998, 2002, Stein, 2005a, Ver Hoef et al., 2004). Let us also mention data-driven aggregation techniques: composite Gaussian process models (Ba and Joseph, 2012), and mixture of experts techniques (see, e.g., Meeds and Osindero, 2006, Rasmussen and Ghahramani, 2002, Tresp, 2001, Yang and Ma, 2011, Yuan and Neubauer, 2009, Yuksel et al., 2012). In the latter framework, the weights are called gating functions and the estimation of the parameters and the inference are usually performed using EM, MCMC, or variational techniques. Weighting methods generally have parameters specifying weighting functions, with an increased need to watch for overfitting phenomena.

### Transformation and composition of models

A first technique for composition of models consists in using a parametric transformation of a GP (Rychlik et al., 1997, Snelson et al., 2004).

Another route is to transform the input domain, using for instance a parametric density (Xiong et al., 2007), or other parametric transformations involving possible dimension reduction (Marmin et al., 2018). Bodin et al. (2020) proposed a framework that uses additional input variables, serving as nuisance parameters, to smooth out some badly behaved data. The practitioner has to specify a prior over the variance of the nuisance parameter and inference is based on MCMC.

Lázaro-Gredilla (2012) takes the step of choosing a GP prior on the output

transform and resorts to variational inference techniques for inference. This type of idea can be viewed as an ancestor of deep Gaussian processes (see, e.g., Bachoc and Lagnoux, 2021, Damianou and Lawrence, 2013, Dunlop et al., 2018, Hebbal et al., 2021, Jakkala, 2021), which stack layers of linear combinations of GPs. The practitioner has to specify a network structure among other parameters and resort to variational inference.

Recently, Picheny et al. (2019) proposed another approach where prediction is made only from pairwise comparisons between data points, relying on the variational framework of ordinal GP regression proposed by Chu and Ghahramani (2005) for the inference.

### 4.1.3 . Contributions and outline

The brief review of the literature above reveals three types of shortcomings in methods that depart from the stationarity hypothesis: 1) they rely on advanced techniques for deriving predictive distributions; or 2) they require the practitioner to choose in advance some key parameters; or 3) they increase the number of parameters with an increased risk of overfitting.

This chapter suggests a method for building models targeting regions of interest specified through function values. The main objective is to obtain global models that exhibit good predictive distributions on a range of interest. In the case of a minimization problem, the range of interest would be the values below a threshold. Outside the range of interest, we accept that the model can be less predictive by relaxing the interpolation constraints. Such a model is presented in Figure 4.3: compared to the situation in Figure 4.2, the model is more predictive in the region where the Steep function takes low values, with expected benefits for the efficiency of Bayesian optimization.

This chapter provides three main contributions. First, we propose a class of goal-oriented GP-based models called *relaxed Gaussian processes* (reGP). Second, we give theoretical and empirical results justifying the method and its use for Bayesian optimization. Finally, to assess the predictivity of reGP, we adopt the formalism of scoring rules (Gneiting and Raftery, 2007) and propose the use of a goal-oriented scoring rule that we call *truncated continuous ranked probability score* (tCRPS), which is designed to assess the predictivity of a model in a range of interest.

The organization is as follows. Section 4.2 briefly recalls the formalism of Gaussian processes and Bayesian optimization (BO). Section 4.3 presents reGP and its theoretical properties. The tCRPS and its use for selecting regions of interest are then presented in Section 4.4. Section 4.5 presents a reGP-based Bayesian optimization algorithm called EGO-R, together with the convergence analysis of this algorithm and a numerical benchmark. Finally, Section 4.6 presents our conclusions and perspectives for future work.

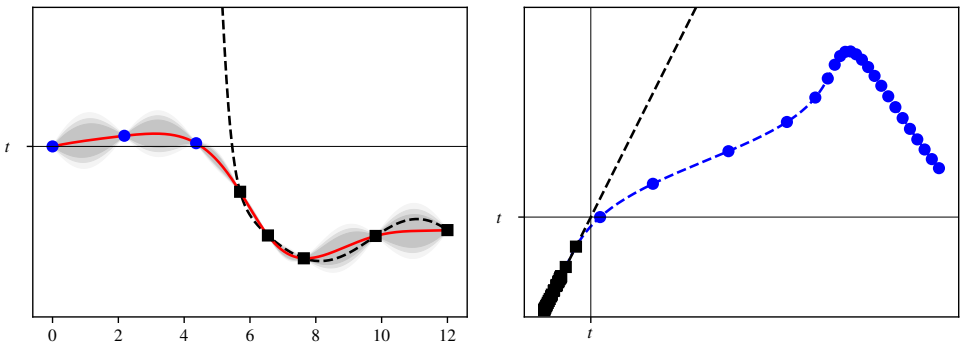



Figure 4.3: Left: prediction of the Steep function with the proposed methodology (black line: relaxation threshold $t$; blue points: relaxed observations). Right: $\mu_n$ versus $f$ (with more observations for illustration purposes). The model interpolates the data below $t$. The blue points are relaxed observations.

## 4.2 . Background and notations

### 4.2.1 . Gaussian process modeling

Consider a real-valued function $f : \mathbb{X} \to \mathbb{R}$, where $\mathbb{X} \subseteq \mathbb{R}^d$, and suppose we want to infer $f$ at a given $x \in \mathbb{X}$ from evaluations of $f$ on a finite set of points $\underline{x}_n = (x_1, \ldots, x_n) \in \mathbb{X}^n$, $n \geq 1$. A standard Bayesian approach to this problem consists in using a GP model $\xi \sim \mathrm{GP}(\mu, k)$ as a prior about $f$, where $\mu : \mathbb{X} \to \mathbb{R}$ is a mean function and $k : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ is a covariance function, which is supposed to be strictly positive-definite in this chapter.

The posterior distribution of $\xi$ given $\underline{Z}_n = (\xi(x_1), \cdots, \xi(x_n))^\mathsf{T}$ is still a Gaussian process, whose mean and covariance functions are given by the standard kriging equations (Matheron, 1971). More precisely:

$$\xi \,|\, \underline{Z}_n \sim \mathrm{GP}(\mu_n, k_n), \tag{4.1}$$

with

$$\mu_n(x) = \mu(x) + k(x, \underline{x}_n) K_n^{-1} (\underline{Z}_n - \mu(\underline{x}_n)) \tag{4.2}$$

and

$$k_n(x, y) = k(x, y) - k(x, \underline{x}_n) K_n^{-1} k(y, \underline{x}_n)^\mathsf{T}, \tag{4.3}$$

and where $\mu(\underline{x}_n) = (\mu(x_1), \ldots, \mu(x_n))^\mathsf{T}$, $k(x, \underline{x}_n) = (k(x, x_1), \ldots, k(x, x_n))$, and $K_n$ is the $n \times n$ matrix with entries $k(x_i, x_j)$. We shall also use the notation $\sigma_n^2(x) = k_n(x, x)$ for the posterior variance, a.k.a. the kriging variance, a.k.a. the squared power function, so that $\xi(x) \,|\, \underline{Z}_n \sim \mathcal{N}\left(\mu_n(x), \sigma_n^2(x)\right)$.

The functions $\mu$ and $k$ control the posterior distribution (4.1) and must be chosen carefully. The standard practice is to select them from data within a parametric family $\{(\mu_\theta, k_\theta),\ \theta \in \Theta\}$. A common approach is to suppose stationarity for the GP, which means choosing a constant mean function $\mu \equiv c \in \mathbb{R}$ and a stationary covariance function $k(x,y) = \sigma^2 r(x-y)$, where $r : \mathbb{R}^d \to \mathbb{R}$ is a stationary correlation function.

A correlation function often recommended in the literature (Stein, 1999) is the (geometrically anisotropic) Matérn correlation function

$$r(h) = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\|h\|_\rho\right)^\nu \mathcal{K}_\nu \left(\sqrt{2\nu}\|h\|_\rho\right), \quad \|h\|_\rho^2 = \sum_{j=1}^{d} \frac{h_{[j]}^2}{\rho_j^2}, \tag{4.4}$$

for $h = (h_{[1]}, \ldots, h_{[d]}) \in \mathbb{R}^d$, and where $\Gamma$ is the Gamma function and $\mathcal{K}_\nu$ is the modified Bessel function of the second kind. The process parameters to be selected in this case are $\theta = (c, \sigma^2, \rho_1, \ldots, \rho_d, \nu) \in \mathbb{R} \times (0,\infty)^{d+2}$ with $\sigma^2$ the process variance, $\rho_i$ the range parameter along the $i$-th dimension, and $\nu$ a regularity parameter controlling the smoothness of the process. Two other standard covariance functions can be recovered for specific values of $\nu$: the exponential covariance function for $\nu = 1/2$ and the squared-exponential covariance function for $\nu \to \infty$.

A variety of techniques for selecting the parameter $\theta$ have been proposed in the literature, but we can safely say that maximum likelihood estimation is the most popular and can be recommended in the case of interpolation (Petit et al., 2021a). It simply consists in minimizing the negative log-likelihood

$$\mathcal{L}(\theta; \underline{Z}_n) = -\log\left(p\left(\underline{Z}_n \,|\, \theta\right)\right) \propto \log\left(\det\left(K_n\right)\right) + \left(\underline{Z}_n - \mu(\underline{x}_n)\right)^\mathsf{T} K_n^{-1} \left(\underline{Z}_n - \mu(\underline{x}_n)\right) + C, \tag{4.5}$$

where $p$ stands for the probability density of $\underline{Z}_n$ and $C$ is a constant. Other methods for selecting the parameters include the restricted maximum likelihood method and leave-one-out strategies (see, e.g., Rasmussen and Williams, 2006, Stein, 1999).

### 4.2.2 . Bayesian optimization

The framework of GPs is well suited to the problem of sequential design of experiments, or active learning. In particular, for minimizing a real-valued function $f$ defined on a compact domain $\mathbb{X}$, the Bayesian approach consists in choosing sequentially evaluation points $X_1, X_2, \ldots \in \mathbb{X}$ using a GP model $\xi$ for $f$, which makes it to possible to build a sampling criterion that represents an expected information gain on the minimum of $f$ when an evaluation is made at a new point. One of the most popular sampling criterion (also called acquisition function) is the *Expected Improvement* (EI) (Jones et al., 1998, Mockus et al., 1978), which can be expressed as

$$\rho_n(x) = \mathsf{E}\left((m_n - \xi(x))_+ \,|\, \underline{Z}_n\right), \tag{4.6}$$

where $m_n = \min(\xi(x_1), \cdots, \xi(x_n))$. The EI criterion corresponds to the expectation of the excursion of $\xi$ below the minimum given $n$ observations, and can be written in closed form:

**Proposition 12** *(Jones et al., 1998, Vazquez and Bect, 2010b) The EI criterion may be written as* $\rho_n(x) = \gamma\big(m_n - \mu_n(x), \sigma_n^2(x)\big)$ *with*

$$\gamma : (z, s) \in \mathbb{R} \times \mathbb{R}_+ \mapsto \begin{cases} \sqrt{s}\phi\left(\frac{z}{\sqrt{s}}\right) + z\Phi\left(\frac{z}{\sqrt{s}}\right) & \text{if } s > 0, \\ \max(z, 0) & \text{if } s = 0, \end{cases}$$

*where* $\phi$ *and* $\Phi$ *stand for the probability density and cumulative distribution functions of the standard Gaussian distribution. Moreover, the function* $\gamma$ *is continuous, verifies* $\gamma(z, s) > 0$ *if* $s > 0$ *and is non-decreasing with respect to* $z$ *and* $s$ *on* $\mathbb{R} \times \mathbb{R}_+$.

When the EI criterion is used for optimization, that is, when the sequence of evaluation points $(X_n)_{n>0}$ of $f$ is chosen using the rule

$$X_{n+1} = \operatorname*{arg\,max}_{x \in \mathbb{X}} \rho_n(x)\,,$$

the resulting algorithm is generally called the Efficient Global Optimization (EGO) algorithm, as proposed by Jones et al. (1998). The EGO algorithm has known convergence properties (Bull, 2011, Vazquez and Bect, 2010b).

A variety of other sampling criteria for the minimization problem can be found in the literature (see, e.g., Frazier et al., 2008, Srinivas et al., 2010, Vazquez and Bect, 2014, Villemonteix et al., 2009), but we shall focus on the EI algorithm in this chapter.

### 4.2.3 . Reproducing kernel Hilbert spaces

Reproducing kernel Hilbert spaces (RKHS, see e.g., Aronszajn, 1950, Berlinet and Thomas-Agnan, 2004) are Hilbert spaces of functions commonly used in the field of approximation theory (see, e.g., Wahba, 1990, Wendland, 2004). A Hilbert space $\mathscr{H}(\mathbb{X})$ of real-valued functions on $\mathbb{X}$ with an inner product $(\cdot,\cdot)_{\mathscr{H}(\mathbb{X})}$ is called an RKHS if it has a reproducing kernel, that is, a function $k\colon \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ such that $k(x, \cdot) \in \mathscr{H}(\mathbb{X})$, and

$$(f, k(x, \cdot))_{\mathscr{H}(\mathbb{X})} = f(x) \tag{4.7}$$

(the reproduction property), for all $x \in \mathbb{X}$ and $f \in \mathscr{H}(\mathbb{X})$. Furthermore, given a (strictly) positive definite covariance function $k$, there exists a unique RKHS admitting $k$ as reproducing kernel.

Given locations $\underline{x}_n = (x_1, \cdots, x_n) \in \mathbb{X}^n$, and corresponding values $\underline{z}_n \in \mathbb{R}^n$, suppose we want to find a function $g \in \mathscr{H}(\mathbb{X})$ such that $g(\underline{x}_n) = (g(x_1), \ldots, g(x_n))^\mathsf{T} = \underline{z}_n$. Then, the minimum-norm interpolation solution is given by the following proposition.

**Proposition 13** *(Kimeldorf and Wahba, 1970) The problem*

$$\begin{cases} \min & \|g\|_{\mathscr{H}(\mathbb{X})}, \\ g \in \mathscr{H}(\mathbb{X}) \\ g(\underline{x}_n) = \underline{z}_n \end{cases} \tag{4.8}$$

*has a unique solution given by* $s_{\underline{z}_n} = k(\cdot, \underline{x}_n) K_n^{-1} \underline{z}_n$.

Observe that the solution $s_{\underline{z}_n}$ is equal to the posterior mean (4.2) when $\mu = 0$.

Moreover, for any $f \in \mathscr{H}(\mathbb{X})$ and $x \in \mathbb{X}$, the reproduction property (4.7) yields the upper bound

$$\left|f(x) - s_{\underline{z}_n}(x)\right| \leq \sigma_n(x)\, \|f\|_{\mathscr{H}(\mathbb{X})}, \tag{4.9}$$

with $\sigma_n(x) = \sqrt{k_n(x,x)}$. Note that $\sigma_n(x)$ is the worst-case error at $x$ for the interpolation of functions in the unit ball of $\mathscr{H}(\mathbb{X})$.

## 4.3 . Relaxed Gaussian process interpolation

### 4.3.1 . Relaxed interpolation

The example in the introduction (see Figures 4.1–4.3) suggests that, in order to gain accuracy over a range of values of interest, it can be beneficial to relax interpolation constraints outside this range. More precisely, the probabilistic model in Figure 4.3 interpolates data lying below a selected threshold $t$, and when data are above $t$, the model only keeps the information that the data exceeds $t$.

In the following, we consider the general setting where relaxation is carried out on a set of the form $R = \bigcup_{j=1}^{J} R_j$, where $R_1, \ldots, R_J \subset \mathbb{R}$ are disjoint closed intervals with non-zero lengths. (The set $R = [t, +\infty)$ was used in the example of Figure 4.3).

As above, we shall write $\underline{x}_n = (x_1, \cdots, x_n) \in \mathbb{X}^n$ for a sequence of locations with corresponding function values $\underline{z}_n = (z_1, \cdots, z_n)^\mathsf{T} \in \mathbb{R}^n$. Then, we introduce the set $C_{R,n} = C_1 \times \cdots \times C_n \subset \mathbb{R}^n$ of relaxed constraints, where

$$\begin{cases} C_i = R_j & \text{if } z_i \in R_j \text{ for some } j, \\ C_i = \{z_i\} & \text{otherwise.} \end{cases} \tag{4.10}$$

Let also

$$\mathscr{H}_{R,n} = \{g \in \mathscr{H}(\mathbb{X}) \mid g(\underline{x}_n) \in C_{R,n}\} \tag{4.11}$$

be the set of relaxed-interpolating functions. The following proposition gives the definition of the minimum-norm relaxed predictor.

**Proposition 14** *The problem*

$$\min_{g \in \mathscr{H}_{R,n}} \|g\|_{\mathscr{H}(\mathbb{X})} \tag{4.12}$$

*has a unique solution given by $s_{\underline{z}_n^\star}$, where $\underline{z}_n^\star$ is the unique solution of the quadratic problem*

$$\operatorname*{arg\,min}_{\underline{z} \in C_{R,n}} \ \underline{z}^\mathsf{T} K_n^{-1} \underline{z}. \tag{4.13}$$

### 4.3.2 . Relaxed Gaussian process interpolation

The main advantage of Gaussian processes is the possibility to obtain not only point predictions but also predictive distributions. However, Proposition 14

only defines a function approximation. We now turn relaxed interpolation into a probabilistic model providing predictive distributions whose mean is not constrained to interpolate data on a given range $R$. The following proposition makes a step in this direction.

**Proposition 15** *Let $\xi \sim \mathrm{GP}(0, k)$, $\underline{x}_n = (x_1, \cdots, x_n) \in \mathbb{X}^n$, $\underline{z}_n \in \mathbb{R}^n$ and $\underline{x}'_m = (x'_1, \cdots, x'_m) \in \mathbb{X}^m$ be a set of locations of interest where predictions should be made. Write $\underline{Z}_n = (\xi(x_1), \ldots, \xi(x_n))^\mathsf{T}$ and $\underline{Z}'_m = (\xi(x'_1), \ldots, \xi(x'_m))^\mathsf{T}$. Then the mode of the probability density function*

$$p\left(\underline{Z}'_m, \underline{Z}_n \,|\, \underline{Z}_n \in C_{R,n}\right) \tag{4.14}$$

*is given by $\left(s_{\underline{z}^\star_n}(\underline{x}'_m),\ \underline{z}^\star_n\right)$.*

In other words, the relaxed interpolation solution of Proposition 14 corresponds to the maximum a posteriori (MAP) estimate under the predictive model (4.14). Conditional distributions with respect to events of the type $\underline{Z}_n \in C_{R,n}$ have been used in Bayesian statistics for dealing with outliers and model misspecifications (see, e.g., Lewis et al., 2021, and references therein). This type of conditional distributions is also encountered for constrained GPs (see, e.g., Da Veiga and Marrel, 2012, López-Lopera et al., 2018, Maatouk and Bay, 2017), when constraints come from expert knowledge.

However, the predictive distribution (4.14) is non-Gaussian since the support of $\underline{Z}_n$ is truncated. In particular, its moments are more expensive to compute than those of a GP (Da Veiga and Marrel, 2012), and sampling requires advanced techniques (e.g., variational, MCMC). Motivated by this observation, we propose instead to build a goal-oriented probabilistic model using the following definition.

**Definition 16 (Relaxed-GP predictive distribution; fixed $\mu$ and $k$)** *Given $\underline{x}_n \in \mathbb{X}^n$, $\underline{z}_n \in \mathbb{R}^n$, and a relaxation set $R$ (finite union of closed intervals), the* relaxed-GP *(reGP) predictive distribution with fixed mean function $\mu$ and covariance function $k$ is defined as the (Gaussian) conditional distribution of $\xi \sim \mathrm{GP}(\mu, k)$ given $\underline{Z}_n = \underline{z}^\star_n$, where $\underline{z}^\star_n$ is given by*

$$\underline{z}^\star_n = \underset{\underline{z} \in C_{R,n}}{\arg\min}\ (\underline{z} - \mu(\underline{x}_n))^\mathsf{T} K_n^{-1} (\underline{z} - \mu(\underline{x}_n)), \tag{4.15}$$

*with $C_{R,n}$ defined by (4.10).*

Observe that (4.15) reduces to (4.13) when $\mu = 0$. Consequently, the mean of the distribution is the predictor $s_{\underline{z}^\star_n}$ from Proposition 14 in this particular case, and is equal to $\mu + s_{\underline{z}^\star_n}$ in general. Moreover, the reGP predictive distribution can be seen as an approximation of (4.14), where $p(\underline{Z}_n \,|\, \underline{Z}_n \in C_{R,n})$ has been replaced by its mode. As discussed earlier, the main advantage of the reGP predictive distribution compared to (4.14) is its reasonable computational burden since it is a GP. Therefore, it makes it possible to use adaptive strategies for the choice of

$R$, as in Section 4.4. Moreover, it also has appealing theoretical approximation properties, as discussed in Section 4.3.3.

As discussed in Section 4.2.1, the standard practice is to select the mean and the covariance functions within a parametric family $\{(\mu_\theta, k_\theta)\,,\ \theta \in \Theta\}$. In this case, we propose to perform the parameter selection and the relaxation jointly. This is formalized by the following definition of relaxed Gaussian process interpolation.

**Definition 17 (Relaxed-GP predictive distribution; estimated parameters)** *Given $\underline{x}_n \in \mathbb{X}^n$, $\underline{z}_n \in \mathbb{R}^n$, a relaxation set $R$ (finite union of closed intervals), and parametric families $(\mu_\theta)$ and $(k_\theta)$ as in Section 4.2.1, the* relaxed-GP *(reGP) predictive distribution with estimated parameters is the (Gaussian) conditional distribution of $\xi \sim \mathrm{GP}(\mu_\theta, k_\theta)$ given $\underline{Z}_n = \underline{z}_n^\star$, where $\underline{z}_n^\star$ and $\theta = \widehat{\theta}_n$ are obtained jointly by minimizing the negative log-likelihood:*

$$\left(\widehat{\theta}_n,\ \underline{z}_n^\star\right) \ = \ \underset{\theta\in\Theta,\ \underline{z}\in C_{R,n}}{\arg\min}\ \mathscr{L}\left(\theta; \underline{z}\right), \tag{4.16}$$

*with $C_{R,n}$ defined by* (4.10)*.*

**Remark 18 (On minimizing** (4.16) **jointly)** *Let $\underline{Z}_{n,1}$ be the values within the range $R$, and $\underline{Z}_{n,0}$ the values in $R^c = \mathbb{R} \setminus R$ that are not relaxed. The negative log-likelihood can be written as*

$$\mathscr{L}\left(\theta; \underline{Z}_n\right) = -\ln\left(p\left(\underline{Z}_{n,0} \,|\, \theta\right)\right) - \ln\left(p\left(\underline{Z}_{n,1} \,|\, \theta, \underline{Z}_{n,0}\right)\right), \tag{4.17}$$

*where the first term is a goodness-of-fit criterion based on the values in $R^c$, and where the second term can mainly be viewed as an imputation term, which "reshapes" the values in $R$ with the information from $\underline{Z}_{n,0}$. (Note also that $\theta$ appears in the second term. When this term is minimized with respect to $\underline{Z}_{n,1}$, it becomes a parameter selection term that promotes the $\theta$s compatible with the excursions in $C_{R,n}$.)*

For illustration, we provide an example of a reGP predictive distribution in Figure 4.4, with an union of two intervals for the relaxation set $R$.

**Remark 19 (Numerical details)** *Minimizing* (4.16) *with respect to $\underline{z}$ falls under the scope of quadratic programming (see, e.g., Nocedal and Wright, 2006b) and could be solved efficiently using dedicated algorithms. This suggests that specific algorithms could be developed for the problem. In this work, we simply use a standard L-BFGS-B solver (Byrd et al., 1995) using the gradient of* (4.16)*.*

### 4.3.3 . Convergence analysis of reGP

In this section, we provide several theoretical results concerning the convergence of the method proposed. This section can be skipped on first reading.

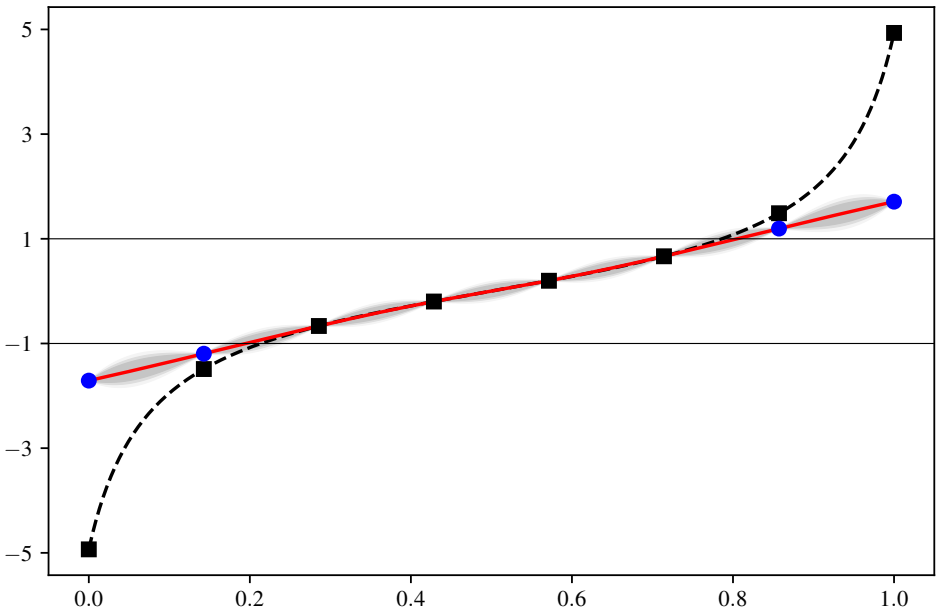



Figure 4.4: An example of reGP predictive distribution with $R = (-\infty, -1] \cup [1, +\infty)$ on a function $f$ represented in dashed black lines. The solid black lines represent the relaxation thresholds. The problem (4.16) was solved only in $\underline{z}$ as the parameters of the (constant) mean and ($\nu = 5/2$ Matérn) covariance functions were held fixed for illustration purposes.

### Known convergence results about interpolation in RKHS

Recall that the fractional-order Sobolev space $W_2^\beta(\mathbb{R}^d)$, with regularity $\beta \geq 0$, is the space of functions on $\mathbb{R}^d$ defined by

$$W_2^\beta(\mathbb{R}^d) = \left\{ g \in \mathbb{L}^2(\mathbb{R}^d), \|g\|_{W_2^\beta(\mathbb{R}^d)} = \left\| \widehat{g} \left(1 + \|\cdot\|^2\right)^{\beta/2} \right\|_{\mathbb{L}^2(\mathbb{R}^d)} < +\infty \right\},$$

where $\widehat{g} \in \mathbb{L}^2(\mathbb{R}^d)$ is the Fourier transform of $g \in \mathbb{L}^2(\mathbb{R}^d)$.

For a given $\mathbb{X} \subset \mathbb{R}^d$, define the Sobolev spaces $W_2^\beta(\mathbb{X}) = \left\{ g_{|\mathbb{X}}, g \in W_2^\beta(\mathbb{R}^d) \right\}$ endowed with the norm

$$\|g\|_{W_2^\beta(\mathbb{X})} = \inf_{\tilde{g} \in W_2^\beta(\mathbb{R}^d), \tilde{g}_{|\mathbb{X}} = g} \|\tilde{g}\|_{W_2^\beta(\mathbb{R}^d)}. \tag{4.18}$$

The following assumption about $\mathbb{X}$ will sometimes be used in this section.

**Assumption 20** *The domain is non-empty, compact, connected, has locally Lipschitz boundary (see, e.g., Adams and Fournier, 2003, Section 4.9), and is equal to the closure of its interior.*

Assumption 20 ensures that the previous definition coincides with other commons definitions, and makes it possible to use well-known results from the field of scattered data approximation, by preventing the existence of cusps. Many common domains—such as hyperrectangles or balls, for instance—verify Assumption 20.

A strictly positive-definite reproducing kernel $k : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ is said to have regularity $\alpha > 0$ if the associated RKHS $\mathscr{H}(\mathbb{X})$ coincides with $W_2^{\alpha+d/2}(\mathbb{X})$ as a function space, with equivalent norms. As such, the Matérn stationary kernels (4.4) have correlation functions $r$ whose Fourier transform verifies (see, e.g., Wendland, 2004, Theorem 6.13)

$$C_1 \left(1 + \|\cdot\|^2\right)^{-\nu-d/2} \leq \widehat{r} \leq C_2 \left(1 + \|\cdot\|^2\right)^{-\nu-d/2}$$

for some $C_2 \geq C_1 > 0$, and have therefore Sobolev regularity $\alpha = \nu$ on $\mathbb{R}^d$ (see, e.g., Wendland, 2004, Corollary 10.13) and consequently also on $\mathbb{X}$, using (4.18) and Lemma 36. Other examples are given by Wendland (2004), for instance.

We now recall a classical convergence result about interpolation in RKHS with evaluation points in a bounded domain. Consider a kernel $k : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$, and let $(x_n)_{n \geq 1} \in \mathbb{X}^{\mathbb{N}}$ be a sequence of distinct points. The following property (a minor reformulation of Theorem 4.1 of Arcangéli et al. (2007)) gives error bounds that depend on the Sobolev regularity of $k$ and the so-called fill distance of $\underline{x}_n \in \mathbb{X}^n$, defined by

$$h_n = \sup_{x \in \mathbb{X}} \min_{1 \leq i \leq n} \|x - x_i\|. \tag{4.19}$$

**Proposition 21** *Let $k$ be a reproducing kernel with regularity $\alpha > 0$. If $\mathbb{X}$ verifies Assumption 20, then*

$$\sup_{x \in \mathbb{X}} \sigma_n(x) \lesssim h_n^{\alpha}, \quad n \geq 1, \tag{4.20}$$

*where $\lesssim$ denotes inequality up to a constant, that does not depend on $(x_n)_{n \geq 1}$.*

Using (4.9) and Proposition 21, this yields the following uniform bound.

**Corollary 22** *Let $k$ be a reproducing kernel with regularity $\alpha > 0$, $\mathscr{H}(\mathbb{X})$ the RKHS generated by $k$, and let $f \in \mathscr{H}(\mathbb{X})$. As above, let $s_{z_n}$ be the solution of (4.8) for $z_n = (f(x_1), \cdots, f(x_n))^{\mathsf{T}}$, $n \geq 1$. If $\mathbb{X}$ verifies Assumption 20, then*

$$\|f - s_{z_n}\|_{\mathbb{L}^{\infty}(\mathbb{X})} \lesssim h_n^{\alpha} \|f\|_{\mathscr{H}(\mathbb{X})}. \tag{4.21}$$

## Convergence results for reGP

Let $k \colon \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ be a continuous strictly positive-definite reproducing kernel. In this section, we consider the zero-mean reGP predictive distribution obtained from $\xi \sim \mathrm{GP}(0, k)$, with relaxed interpolation constraints on a union $R = \bigcup_{j=1}^{q} R_j$ of disjoints closed intervals $R_j$ with non-zero length. Let $\mathscr{H}(\mathbb{X})$ be the RKHS attached to $k$, $f \in \mathscr{H}(\mathbb{X})$, and consider a sequence $(x_n)_{n \geq 1} \in \mathbb{X}^{\mathbb{N}}$ of distinct points. Furthermore, define (the) regions $\mathbb{X}_j = \{x \in \mathbb{X},\ f(x) \in R_j\}$ for $1 \leq j \leq q$ and $\mathbb{X}_0 = \mathbb{X} \setminus \bigcup_{j \geq 1} \mathbb{X}_j$. We give results about the limit of the sequence of reGP predictive distributions that suggest an improved fit in $\mathbb{X}_0$.

Let $s_{R,n} = s_{z_n^{\star}}$ be the relaxed predictor from Proposition 14 based on $(x_1, \ldots, x_n)$ and $(f(x_1), \ldots, f(x_n))^{\mathsf{T}}$, $n \geq 1$. The following proposition establishes the limit behavior of the sequence $(s_{R,n})_{n \geq 1}$.

**Proposition 23** *Let $\mathbb{U} \subset \mathbb{X}$ and let $\mathscr{H}_{R,\mathbb{U}}$ denote the set of functions $g \in \mathscr{H}(\mathbb{X})$ such that, for all $x \in \mathbb{U}$,*

$$\begin{cases} g(x) \in R_j & \text{if } f(x) \in R_j \text{ for some } j, \\ g(x) = f(x) & \text{otherwise.} \end{cases} \tag{4.22}$$

*Then the problem*

$$\min_{g \in \mathscr{H}_{R,\mathbb{U}}} \|g\|_{\mathscr{H}(\mathbb{X})} \tag{4.23}$$

*has a unique solution denoted by $s_{R,\mathbb{U}}$. Moreover,*

$$s_{R,n} \xrightarrow{\mathscr{H}(\mathbb{X})} s_{R,\mathbb{U}}, \tag{4.24}$$

*with $\mathbb{U}$ the closure of $\{x_n\}$.*

In particular, when $\{x_n\}$ is dense in $\mathbb{X}$, then $\mathbb{U} = \mathbb{X}$ and $(s_{R,n})_{n \geq 1}$ converges to $s_{R,\mathbb{X}}$, which is the minimal-norm element of the set $\mathscr{H}_{R,\mathbb{X}}$.

The next proposition tells us that the interpolation error on $\mathbb{X}_0$ can be bounded by a term that depends on the norm of $s_{R,\mathbb{X}}$.

**Proposition 24** *For any $x \in \mathbb{X}_0$ and $n \geq 1$,*

$$|f(x) - s_{R,n}(x)| \leq 2\sigma_{n,0}(x)\|s_{R,\mathbb{X}}\|_{\mathscr{H}(\mathbb{X})}, \tag{4.25}$$

*where $\sigma_{n,0}$ is the power function obtained using only points in $\mathbb{X}_0$ for predictions.*

This yields the following error bounds when the design is dense.

**Proposition 25** *Suppose that $\{x_n\}$ is dense and that $k$ has regularity $\alpha > 0$. Let $B \subset \mathbb{X}_0$ verify Assumption 20. Then, for all $n \geq 1$,*

$$\|f - s_{R,n}\|_{\mathbb{L}^\infty(B)} \lesssim h_n^\alpha \|s_{R,\mathbb{X}}\|_{\mathscr{H}(\mathbb{X})}. \tag{4.26}$$

*Let $d(y, A)$ be the distance of $y \in \mathbb{R}$ to $A \subset \mathbb{R}$. For $j \geq 1$, $x \in \mathbb{X}_j$, and for all $n \geq 1$:*

$$d(s_{R,n}(x), R_j) \lesssim h_n^\alpha \|s_{R,\mathbb{X}}\|_{\mathscr{H}(\mathbb{X})} \quad \text{if } \alpha < 1, \tag{4.27}$$

$$d(s_{R,n}(x), R_j) \lesssim \sqrt{(|\ln(h_n)| + 1)}\, h_n \|s_{R,\mathbb{X}}\|_{\mathscr{H}(\mathbb{X})} \quad \text{if } \alpha = 1, \tag{4.28}$$

*and*

$$d(s_{R,n}(x), R_j) \lesssim h_n \|s_{R,\mathbb{X}}\|_{\mathscr{H}(\mathbb{X})} \quad \text{if } \alpha > 1, \tag{4.29}$$

*where $\lesssim$ denotes inequality up to a constant, that does not depend on $f$, $n$, $x$ or $(x_n)$.*

Finally, we investigate the following question: how large can be the norm of $f$ compared to the approximation $\|s_{R,\mathbb{X}}\|_{\mathscr{H}(\mathbb{X})}$?

**Proposition 26** *Suppose that $k$ has regularity $\alpha > 0$ and that there exists some $j \geq 1$ such that $\mathbb{X}_j$ has a non-empty interior. We have*

$$\sup_{g \in \mathscr{H}_{R,\mathbb{X}}} \|g\|_{\mathscr{H}(\mathbb{X})} = +\infty, \tag{4.30}$$

*with $\mathscr{H}_{R,\mathbb{X}}$ given by* (4.22) *for $f \in \mathscr{H}(\mathbb{X})$.*

This result shows that the norm reduction obtained by approximating $f$ with relaxed interpolation constraints can therefore be arbitrarily high in the finite-smoothness case. A stronger version of Proposition 26 for the special case where $R = [t, +\infty)$ can be derived, and shows that

$$\sup_{g \in \mathscr{H}_{R,\mathbb{X}}} \|g\|_{\mathbb{L}^\infty(\mathbb{X})} = +\infty.$$

Overall, no matter the element of $\mathscr{H}_{R,\mathbb{X}}$ at hand, reGP converges to a function $s_{R,\mathbb{X}}$ which: coincides with $f$ on $\mathbb{X}_0$, verifies $f(x) \in R_j \Leftrightarrow s_{R,\mathbb{X}}(x) \in R_j$ for all $x \in \mathbb{X}$, and is "nicer" than $f$ in the sense of $\|\cdot\|_{\mathscr{H}(\mathbb{X})}$. Furthermore, reGP yields error bounds carrying the norm of $s_{R,\mathbb{X}}$, which can be arbitrarily smaller than the norm of $f$ in the case of a finite-smoothness covariance function.

**Remark 27** *Note that $\sigma_n \leq \sigma_{n,0}$ due to the standard projection interpretation. Empirical and theoretical results about the* screening effect *(see, e.g., Bao et al., 2020, Stein, 2011), suggests that $\sigma_n \simeq \sigma_{n,0}$, if $k$ has smoothness $\alpha > 0$. In this case, observe that—no matter the element of $\mathscr{H}_{R,\mathbb{X}}$ at hand—the bound* (4.25) *is larger by only a small factor compared to* (4.9) *with $f = s_{R,\mathbb{X}}$. (However, to the best of our knowledge, no result exists concerning the screening effect for arbitrary designs.)*

**Remark 28** *The equality* (4.30) *does not hold in general for infinitely smooth covariance functions. For instance, Steinwart et al. (2006, Corollary 3.9) show that $\mathscr{H}_{R,\mathbb{X}} = \{f\}$ if the interior of $\mathbb{X}_0$ is not empty and $k$ is the squared-exponential covariance function (i.e.* (4.4)*, with $\nu \to \infty$).*

## 4.4 . Choice of the relaxation set

### 4.4.1 . Towards goal-oriented cross-validation

The framework of reGP makes it possible to predict a function $f$ from point evaluations of $f$. Suppose we are specifically interested in obtaining good predictive distributions in a range $Q \subset \mathbb{R}$ of function values, and accept degraded predictions outside this range. To achieve this goal, the idea of reGP is to relax interpolation constraints. Naturally, it makes sense to relax interpolation constraints outside the range $Q$ but it could happen that relaxing interpolation constraints does not improve predictive distributions on $Q$. Therefore, the question arises as to how to automatically select a range $R$ in $\mathbb{R} \setminus Q$, on which interpolation constraints should be relaxed.

In the following, we put $R^{(0)} = \mathbb{R} \setminus Q$, and we view the relaxation set $R$ as a parameter of the reGP model, which has to be chosen in $R^{(0)}$ along with the parameters $\theta$ of the underlying GP $\xi$. A first idea for the selection of $R$ is to rely on the standard leave-one-out cross-validation approach to select the parameters of a GP (Dubrule, 1983, Rasmussen and Williams, 2006, Zhang and Wang, 2010). Using the formalism of *scoring rules* (see, e.g., Gneiting and Raftery, 2007, Petit et al., 2021a), selecting parameters by a leave-one-out approach amounts to minimizing a selection criterion written as

$$J_n(R) = \frac{1}{n} \sum_{i=1}^{n} S\left(P_{R,n,-i}, f(x_i)\right), \tag{4.31}$$

where $P_{R,n,-i}$ is the reGP predictive distribution with the observations $z_{n,-i} = (z_1, \ldots, z_{i-1}, z_{i+1}, \ldots, z_n)$ and the relaxation set $R$. The function $S$ in (4.31) is a scoring rule, that is, a function $S : \mathscr{P} \times \mathbb{R} \to \mathbb{R} \cup \{-\infty, +\infty\}$, acting on a class $\mathscr{P}$ of probability distributions on $\mathbb{R}$, such that $S(P, z)$ assigns a loss for choosing a predictive distribution $P \in \mathscr{P}$, while observing $z \in \mathbb{R}$. Scoring rules make it possible to quantify the quality of probabilistic predictions.

Since the user is not specifically interested in good predictive distributions in $R^{(0)}$, validating the model on $R^{(0)}$ should not be a primary focus. However, simply restricting the sum (4.31) by removing indices $i$ such that $f(x_i) \in R^{(0)}$ would make it impossible to assess if the model is good at predicting that $f(x) \in R^{(0)}$ for a given $x \in \mathbb{X}$. For instance, in the case of minimization, with $Q = (-\infty, t^{(0)})$ and $R^{(0)} = [t^{(0)}, \infty)$, it is important to identify the regions corresponding to $f$ being above $t^{(0)}$, even if we are not interested in accurate predictions above $t^{(0)}$, because we expect that an optimization algorithm should avoid the exploration of these regions.

In the next section, we propose instead to keep the whole leave-one-out sum (4.31), but to choose a scoring rule $S$ that serves our goal-oriented approach.

### 4.4.2 . Truncated continuous ranked probability score

An appealing class of scoring rules for goal-oriented predictive distributions is the class of weighted scoring rules for binary predictors (Gneiting and Raftery, 2007, Matheson and Winkler, 1976), which may be written as

$$S(P, z) = \int_{-\infty}^{+\infty} s(F_P(u), \mathbb{1}_{z \leq u}) \, \mu(\mathrm{d}u), \tag{4.32}$$

where $s\colon [0,1] \times \{0,1\} \to \mathbb{R} \cup \{-\infty, +\infty\}$ is a scoring rule for binary predictors, and $\mu$ is a Borel measure on $\mathbb{R}$. A well-known instance of (4.32) is the continuous ranked probability score (Gneiting et al., 2005) written as

$$S^{\mathrm{CRPS}}(P, z) = \int_{-\infty}^{+\infty} (F_P(u) - \mathbb{1}_{z \leq u})^2 \, \mathrm{d}u,$$

which is obtained by choosing the Brier score for $s$ and the Lebesgue measure for $\mu$.

For the case where we are specifically interested in obtaining good predictive distributions in a range of interest $Q \subset \mathbb{R}$, we propose to use the following scoring rule, which we call *truncated continuous ranked probability score* (tCRPS):

$$S_Q^{\mathrm{tCRPS}}(P, z) = \int_Q (F_P(u) - \mathbb{1}_{z \leq u})^2 \, \mathrm{d}u. \tag{4.33}$$

This scoring rule, proposed by Lerch and Thorarinsdottir (2013) in a different context, reduces to $S^{\mathrm{CRPS}}$ when $Q = \mathbb{R}$. It can be seen as a special case of the weighted CRPS (Gneiting and Ranjan, 2011, Gneiting and Raftery, 2007, Matheson and Winkler, 1976), in which the indicator function $\mathbb{1}_Q$ plays the role of the weight function—in other words, the measure $\mu$ in (4.32) has density $\mathbb{1}_Q$ with respect to Lebesgue's measure.

Consider for instance the case $Q = \left(-\infty, t^{(0)}\right)$:

$$S_Q^{\mathrm{tCRPS}}(P, z) = \int_{-\infty}^{t^{(0)}} (F_P(u) - \mathbb{1}_{z \leq u})^2 \, \mathrm{d}u. \tag{4.34}$$

Note that, in this case, $S_Q^{\text{tCRPS}}(P, z)$ does not depend on the specific value of $z$ when $z \geq t^{(0)}$. This scoring rule is thus well suited to the problem of measuring the performance of a predictive distribution in such a way as to fully assess the goodness-of-fit of the distribution when the true value is below a threshold, and only ask that the support of the predictive distribution is concentrated above the threshold when the true value is above the threshold.

We provide in Appendix 4.7 some properties of the scoring rule (4.33) and closed-form expressions for the case where $Q$ is an interval (or a finite union of intervals) and $P$ is Gaussian. To the best of our knowledge, these expressions are new.

### 4.4.3 . Choosing the relaxation set using the tCRPS scoring rule

Given a range of interest $Q$, the tCRPS scoring rule makes it possible to derive a goal-oriented leave-one-out selection criterion, that we shall call the LOO-tCRPS criterion:

$$J_n(R) = \frac{1}{n}\sum_{i=1}^{n} S_Q^{\text{tCRPS}}\left(P_{R,n,-i}, f(x_i)\right) . \tag{4.35}$$

Using (4.35), we suggest the following procedure to select a reGP model. First, choose a sequence of nested candidate relaxation sets $R^{(0)} \supset R^{(1)} \supset \cdots \supset R^{(G)} = \emptyset$. The next step is the computation of $J_n(R^{(g)})$, $g = 0, \ldots, G$, which involves the predictive distributions $P_{R^{(g)},n,-i}$.

In principle, (4.16) should be solved again each time a data point $(x_i, z_i)$ is removed, to obtain a pair $(\widehat{\theta}_{n,-i}^{(g)}, \underline{z}_{n,-i}^{(g)})$ and then the corresponding reGP distribution $P_{R^{(g)},n,-i}$. To alleviate computational cost, a simple idea is to rely on the fast leave-one-out formulas (Dubrule, 1983) for Gaussian processes: for each set $R^{(g)}$, solve (4.16) to obtain $\widehat{\theta}_n^{(g)}$ and $\underline{z}_n^{(g)} = (z_1^{(g)}, \ldots, z_n^{(g)})^{\mathsf{T}}$, and then compute the conditional distributions $\xi(x_i) \mid \{\xi(x_j) = z_j^{(g)}, j \neq i\}$, where $\xi \sim \text{GP}(\mu, k)$, and where $\mu$ and $k$ have parameter $\widehat{\theta}_n^{(g)}$, using the fast leave-one-out formulas. By doing so, we neglect the difference between $\widehat{\theta}_{n,-i}^{(g)}$ and $\widehat{\theta}_n^{(g)}$ and the difference between $\underline{z}_{n,-i}^{(g)}$ and the vector $(z_1^{(g)}, \ldots, z_{i-1}^{(g)}, z_{i+1}^{(g)}, \ldots, z_n^{(g)})^{\mathsf{T}}$.

The procedure ends by choosing the relaxation set $R^{(g)}$ that achieves the best LOO-tCRPS value.

Figure 4.5 illustrates the selection of the relaxation set used in Figure 4.3.

### 4.4.4 . An example for the estimation of an excursion set

We illustrate the method on the problem of estimating an excursion set $\{x \in \mathbb{X}, f(x) \leq 0\}$. We consider the g10-RR optimization problem from Section 1.6, and focus on the constraint $c_6 \leq 0$. Finding solutions satisfying the $c_6 \leq 0$ constraint using a GP model is difficult, probably because the values of $c_6$ are very bi-modal, as illustrated in Figure 4.6. However, Feliot et al. (2017) found that the difficulty could be overcome by performing an ad-hoc monotonic transformation $z \mapsto z^{\alpha}$, with $\alpha = 7$, on the constraint.

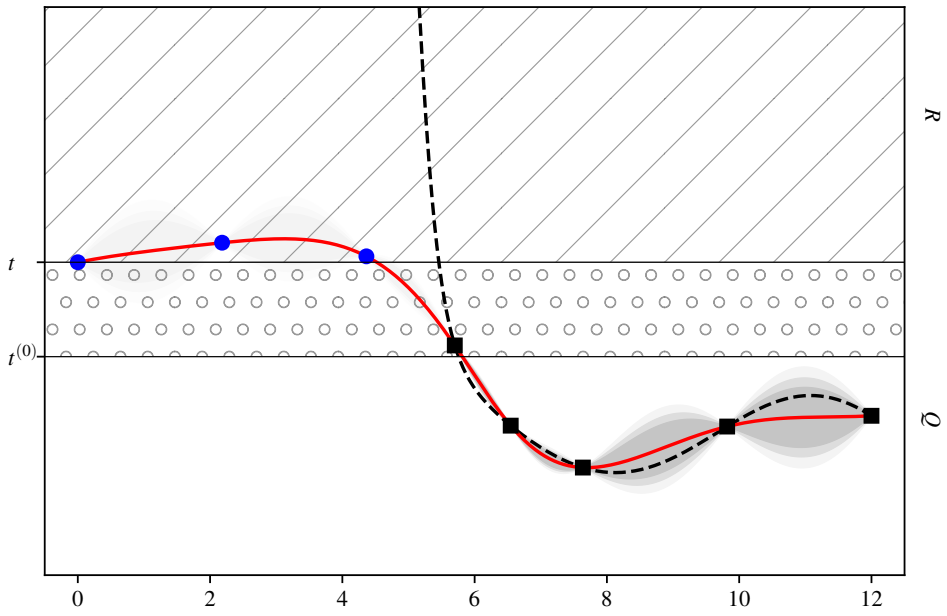



Figure 4.5: Illustration of the choice of a relaxation set. The range of interest $Q$ is determined by the threshold $t^{(0)}$. The relaxation set $R$ corresponding to the region above $t$ has been obtained by the procedure described in Section 4.4.3.

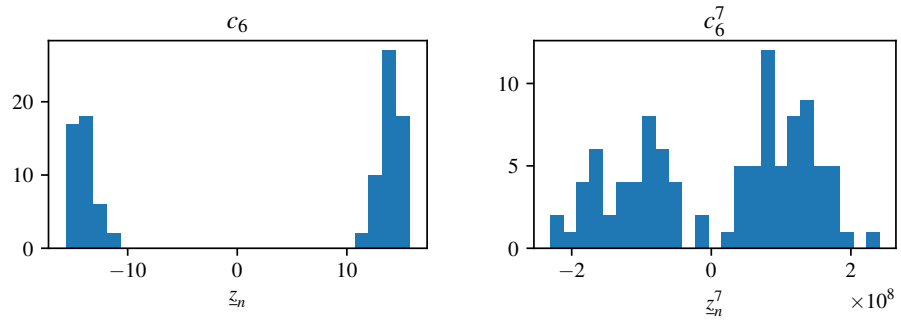



Figure 4.6: Left: Histogram of the values of the function $c_6$ from the g10-RR problem. Right: Same illustration but for the function $c_6^\alpha$, with $\alpha = 7$. The histograms are obtained from the values of the functions on a space-filling design of size $n = 100$. On the left, the values are very separated and concentrated on two modes, yielding a function close to a piecewise constant function. After transformation, the phenomenon is mitigated.

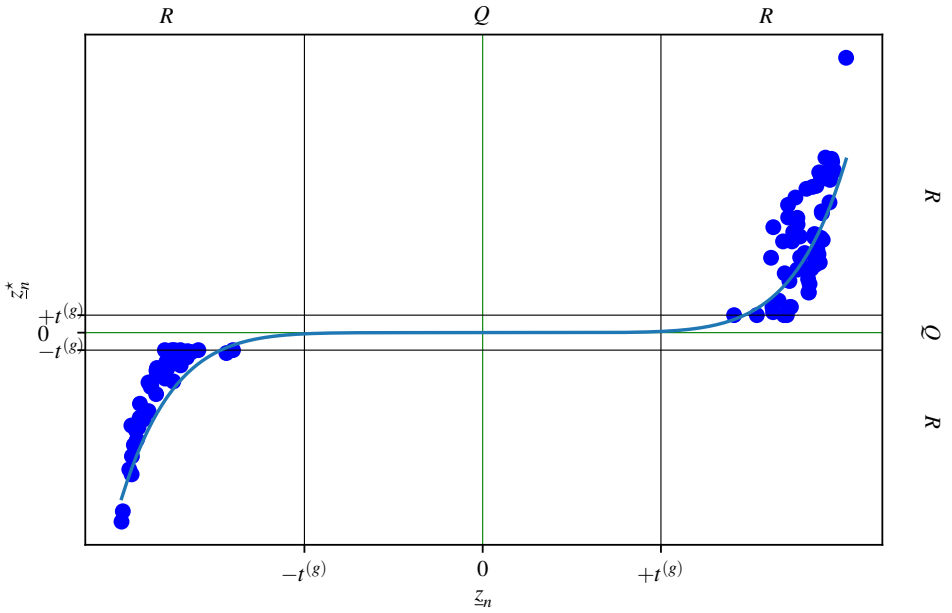



Figure 4.7: A reGP fit of $c_6$, where the relaxation thresholds have been selected by LOO-tCRPS. The observations $z_n$ are shown on the $x$-axis, whereas the "relaxed" observations $z_n^\star$ are represented on the $y$-axis. Moreover, the green lines represent the value zero, and the brown lines represent $\pm\, t^{(g)}$, with $t^{(g)}$ chosen to be $t^{(0)}$ by the LOO-tCRPS. Finally, the blue line shows a best fit by $z \mapsto z^7$.

The estimation of an excursion set $\{f \leq 0\}$ involves capturing precisely the behavior of $f$ around zero. Thus, we define a range of interest $Q = (-t^{(0)}, t^{(0)})$ centered on zero, with $t^{(0)}$ sufficiently small (note that there may be no data in $Q$). Then, we consider relaxation range candidates $R^{(g)} = (-\infty, -t^{(g)}] \cup [t^{(g)}, +\infty)$ with a sequence of thresholds $t^{(0)} < \cdots < t^{(G)} = +\infty$, and we select $t^{(g)}$ by minimizing the LOO-tCRPS as described in the previous section.

In the case of the $c_6$ constraint and a small value for $t^{(0)}$ such that there is no data in $Q$, the results are presented in Figure 4.7. The LOO-tCRPS chooses $t^{(g)} = t^{(0)}$, so the reGP predictive distributions use only the information of being above or below zero. Moreover, observe that the corresponding transformation after relaxation bears resemblance to the transformation $z \mapsto z^\alpha$ proposed by Feliot et al. (2017). If we apply the reGP framework on the transformed function $c_6^\alpha$ (details omitted for brevity), we find that the LOO-tCRPS chooses a large $t^{(g)}$ such that the interpolation constraints are relaxed for a few observations only.

## 4.5 . Application to Bayesian optimization

### 4.5.1 . Efficient global optimization with relaxation

The first motivation for introducing reGP models is Bayesian optimization, where obtaining good predictive distributions over ranges corresponding to optimal values is a key issue. In this chapter, we focus more specifically on the minimization problem

$$\min_{x \in \mathbb{X}} f(x)\,, \tag{4.36}$$

where $f$ is a real-valued function defined on a compact set $\mathbb{X} \subset \mathbb{R}^d$, but the methodology can be generalized to constrained and/or multi-objective formulations.

Given $f$, our objective is to construct a sequence of evaluation points $X_1, X_2 \ldots \in \mathbb{X}$ by choosing each point $X_{n+1}$ as the maximizer of the expected improvement criterion (4.6) computed with respect to the reGP predictive distribution, with a relaxation set $R_n = [t_n, +\infty)$. More precisely, the sequence $(X_n)$ is constructed sequentially using the rule

$$X_{n+1} \;=\; \operatorname*{arg\,max}_{x \in \mathbb{X}} \; \mathsf{E}_n\left((m_n - \xi(x))_+\right)\,, \tag{4.37}$$

where $m_n = f(X_1) \wedge \cdots \wedge f(X_n)$, and $\mathsf{E}_n$ is the expectation under the reGP predictive distribution with relaxation set $R_n$ and data $\underline{z}_n = (f(X_1), \ldots, f(X_n))^{\mathsf{T}}$.

As in Section 4.4.3, the relaxation threshold $t_n$ at iteration $n$ is chosen using the LOO-tCRPS criterion (4.35) among candidates values

$$t_n^{(0)} < t_n^{(1)} < \cdots < t_n^{(G)}\,, \tag{4.38}$$

where $t_n^{(0)}$ is the validation threshold. In the following, the optimization method just described will be called efficient global optimization with relaxation (EGO-R), in reference to the EGO name proposed by Jones et al. (1998).

Implementation specifics—including heuristics for choosing $t_n^{(0)}$—are detailed in Section 4.5.3. In the next section, we show that using the EI criterion with a reGP model yields a convergent algorithm.

### 4.5.2 . Convergence of EGO-R with fixed parameters and varying threshold

In this section, we extend the result of Vazquez and Bect (2010b) and show the convergence of the EGO-R algorithm, in the case where the predictive distributions derive from a zero-mean Gaussian process with fixed covariance function.

We suppose that $\mathbb{X}$ is a compact domain and that $k : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ is continuous, strictly positive-definite, and has the NEB (no-empty ball) property (Vazquez and Bect, 2010b), which says that the posterior variance cannot go to zero at a given point if there is no evaluation points in a ball centered on this point. In other words, the NEB property requires that the posterior variance $\sigma_n^2(x)$ at $x \in \mathbb{X}$ remains bounded away from zero for any $x$ not in the closure of the sequence of points $(X_n)$ evaluated by the optimization algorithm. A stationary covariance function with

smoothness $\alpha > 0$ verifies the NEB property (Vazquez and Bect, 2010b), whereas the squared-exponential covariance function does not (Vazquez and Bect, 2010a).

**Proposition 29** *Let $k : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ be a continuous strictly positive-definite covariance function that verifies the NEB property, $\mathscr{H}(\mathbb{X})$ the corresponding RKHS and $f \in \mathscr{H}(\mathbb{X})$. Let $n_0 > 0$. Let $(X_n)_{n\geq 1}$ be a sequence in $\mathbb{X}$ such that, for each $n \geq n_0$, $X_{n+1}$ is obtained by* (4.37) *with $t_n > m_n$. Then the sequence $(X_n)_{n\geq 1}$ is dense in $\mathbb{X}$.*

Observe that Proposition 29 implies the convergence of EGO-R with a fixed threshold $t > \min_{i\leq n_0} f(X_i)$. In this case, the theoretical insights from Section 4.3.3 suggest a faster convergence might be achieved due to the improved error estimates (4.25) and (4.26) in a neighborhood of the global minimum.

The convergence of EGO-R also holds in the case of a varying relaxation set $R_n = [t_n, +\infty)$, with $t_n > m_n$, and in particular when $t_n$ is selected at each step using the LOO-tCRPS criterion (4.35) with a validation threshold $t_n^{(0)} > m_n$. In this case, the norm term in (4.25) gets smaller if $(t_n)_{n\geq 1}$ is decreasing.

### 4.5.3 . Optimization benchmark

In this section, we run numerical experiments to demonstrate the interest of using EGO-R instead of EGO for minimization problems.

## Methodology

In practice, we must choose the sequence of thresholds (4.38). The validation threshold $t_n^{(0)}$ should be set above $m_n$ to ensure there is enough data to carry out the validation. We propose two different heuristics: a) a *constant* heuristic, where $t_n^{(0)}$ is kept constant through the iterations and set to an empirical quantile of an initial data set constructed before EGO-R is run, and b) a *concentration* heuristic, where $t_n^{(0)}$ corresponds to an empirical quantile of $z_n$.

In the case of the constant heuristic, we set $t_n^{(0)}$ to the $\alpha$-quantile of the function values on an initial design, which is typically built to fill $\mathbb{X}$ as evenly as possible with, e.g., maximin Latin hypercube sampling (McKay et al., 2000). The numerical experiments were conducted with $\alpha = 0.25$ in this chapter.

In the case of the concentration heuristic, we consider an $\alpha$-quantile of the values of $f$ at the points visited by the algorithm (again with $\alpha = 0.25$). As the optimization algorithm makes progress, the evaluations will likely concentrate around the global minimum. Thus, $t_n^{(0)}$ will get closer to the minimum value, and the range $Q_n = (-\infty, t_n^{(0)})$ of validation values will get smaller. Besides, since we expect better predictive distributions in this range, a better convergence may be obtained.

Both heuristics can be justified by the idealized setting of the convergence result from the previous section. Proposing alternative adaptive strategies to the

concentration heuristic, or more generally conducting a theoretical study on the performance of such adaptive strategies, is out of the scope of this chapter.

For a given $t_n^{(0)}$, the candidate relaxation thresholds $t_n^{(g)}$, $g = 1, \dots, G$, are chosen so that $t_n^{(g)} - m_n$ ranges logarithmically from $t_n^{(0)} - m_n$ to $\max f(X_i) - m_n$ (with $G = 10$, in the experiments below).

To assess the performances of EGO-R with the two heuristics for choosing $t_n^{(0)}$, we compare them to the standard EGO algorithm. For all three algorithms, we use a first initial maximin (over 50 random repetitions) Latin hypercube design of size $n_0 = 3d$, and we consider GPs with constant mean and a Matérn covariance function with regularity $\nu = 5/2$. The parameters are estimated using (4.5) for EGO, while EGO-R uses (4.16) for a relaxation set $[t_n, +\infty)$ selected by LOO-tCRPS with $t_n^{(0)}$. The maximization of the sampling criteria (4.6) and (4.37) is carried out using a sequential Monte Carlo approach (Benassi et al., 2012, Feliot et al., 2017).

For reference, we also run the Dual simulated Annealing algorithm (inspired by Xiang et al. (1997)) from SciPy (Virtanen et al., 2020), with the default settings and with a random initialization.

The optimization algorithms are tested against a benchmark of test functions from Surjanovic and Bingham (2013) summarized in Table 4.1, with $n_{\text{rep}} = 30$ random repetitions, and a budget of $n_{\text{tot}} = 300$ evaluations for each repetition. Note that the randomness is caused by the designs and the SMC algorithm.

This benchmark is partly inspired by Jones et al. (1998) and Merrill et al. (2021). In particular, we also use a log-version of the Goldstein-Price function as Jones et al. (1998).

To evaluate the algorithms we use, for each test function, several targets defined as spatial quantiles of the function and estimated with a subset simulation algorithm (see, e.g., Au and Beck, 2001). Then, the performances of the algorithms are evaluated using the fractions of runs that manage to reach the targets and the average numbers of evaluations to reach the targets (with unsuccessful runs counted as $n_{\text{tot}}$).

## Findings

The full set of results is provided in Appendix 4.9. In Figure 4.8, we present a representative subset of these results.

First, observe in Figure 4.8 that the EGO-R methods can be considerably helpful and can outperform EGO largely on functions that are difficult to model with stationary GPs, such as Goldstein-Price, Perm $(10)$, and Beale.

Observe also that the EGO-R methods have about the same performance as EGO on functions that are easy to model with stationary GPs. This is the case of the Log-Goldstein-Price and the Branin functions, for which the LOO-tCRPS criterion for choosing the relaxation set detects that the larger values help predict

Table 4.1: Optimization benchmark. The acronym G-P stands for Goldstein-Price.

| Problem | $d$ |
|---|---|
| Branin | $2$ |
| Six-hump Camel | $2$ |
| Three-hump Camel | $2$ |
| Hartman | $3, 6$ |
| Ackley | $4, 6, 10$ |
| Rosenbrock | $4, 6, 10$ |
| Shekel | $5, 7, 10$ |
| Goldstein-Price | $2$ |
| Log-Goldstein-Price | $2$ |
| Cross-in-Tray | $2$ |
| Beale | $2$ |
| Dixon-Price | $4, 6, 10$ |
| Perm | $4, 6, 10$ |
| Michalewicz | $4, 6, 10$ |
| Zakharov | $4, 6, 10$ |

near the minimum, and that no relaxation is needed as a result.

Furthermore, it is instructive to compare the performances of the EGO-R algorithms on the Goldstein-Price function on the one hand, with the performance of the EGO algorithm on Log-Goldstein-Price function on the other hand. Using reGP modeling enables to perform as well as with the logarithmic transform, but in an automatic way. This is also illustrated by Figure 4.9, which shows that the (non-parametric) transform learned by reGP resembles a logarithmic transform.

Finally, observe that the constant heuristic performs as well as EGO on Ackley $(10)$, whereas the concentration heuristic lags behind. A closer look at the results for this function shows that the concentration heuristic get sometimes stuck in a local minimum. We explain this by the fact that the reGP model with the concentration heuristic can become very predictive in a small region around the local minimum, and underestimate the function variations elsewhere (the variance of the predictive distributions above $t_n^{(0)}$ are too small, and the optimization algorithm does not sufficiently explore unknown regions). To this regard, the constant heuristic is probably more conservative. Overall, taking the results from Appendix 4.9 into account, the concentration heuristic appears to be more (resp. less) efficient than the constant heuristic when there are few (resp. many) local minima.

## 4.6 . Conclusion

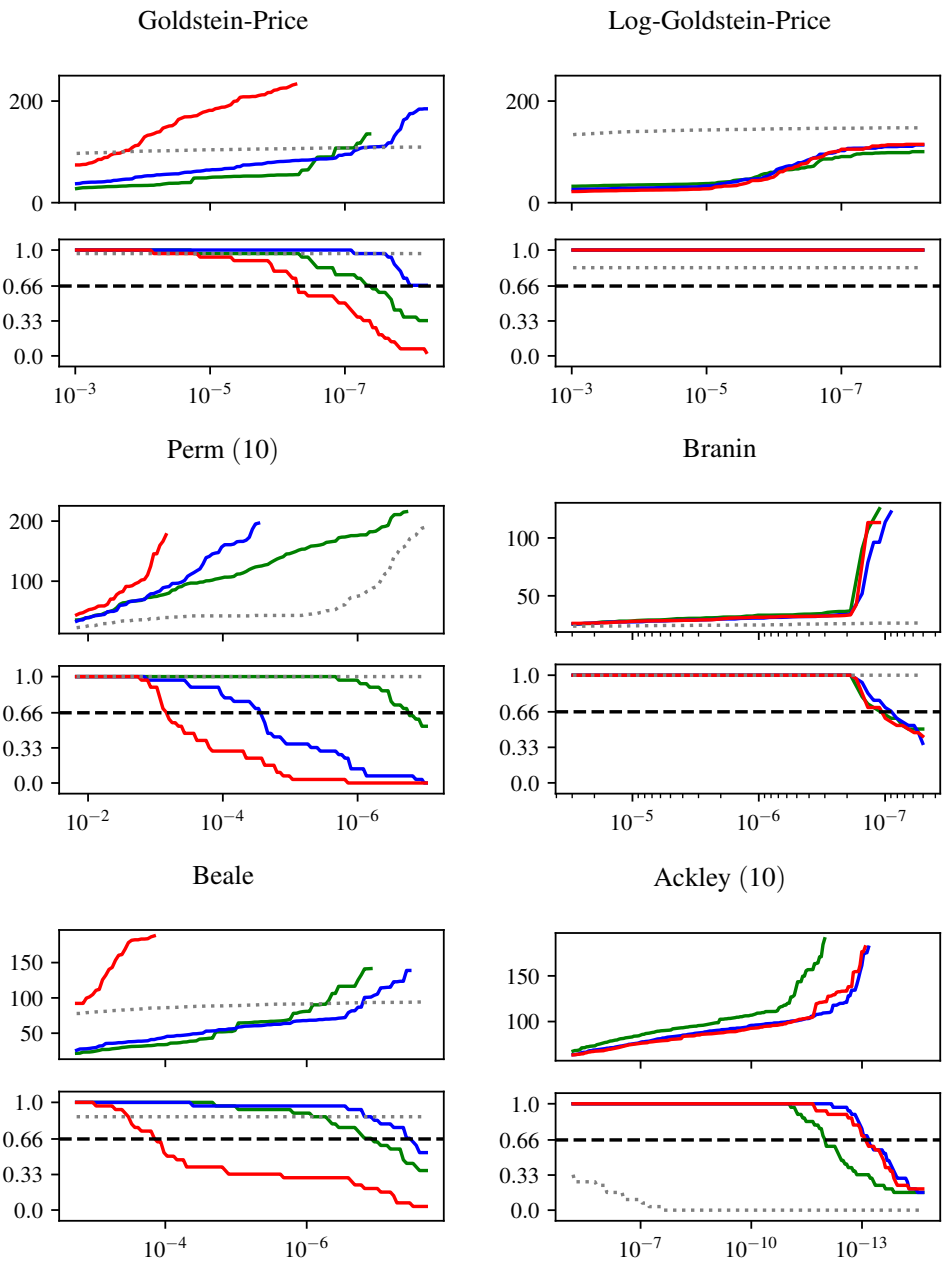



Figure 4.8: For each case, the top plot is the average number of iterations to reach the spatial quantile, and the bottom plot is the proportion of successful runs. Both are represented versus the level of the spatial quantile. The gray dotted line stands for the Dual simulated Annealing algorithm, the red line for the standard EGO algorithm, and the blue and green lines for EGO-R, with the "Constant" and "Concentration" heuristics. The results are shown until the success rates of the three previous methods fall below the 66% threshold materialized by the black dashed line.

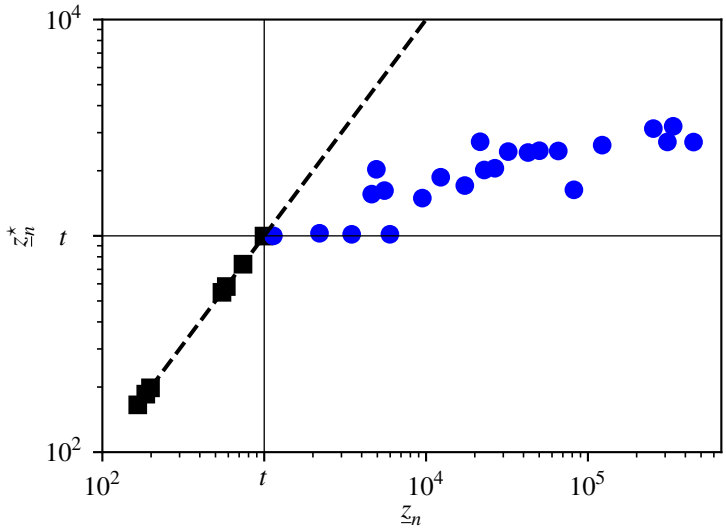



Figure 4.9: A reGP fit to the Goldstein-Price function with $n = 30$ points, with $R = [10^3, +\infty)$. The observations $z_n$ are shown on the $x$-axis, whereas the relaxed observations $z_n^\star$ are represented on the $y$-axis.

This chapter presents a new technique called reGP to build predictive distributions for a function observed on a sequence of points. This technique can be applied when a user wants good predictive distributions in a range of function values, for example below a given threshold, and accepts degraded predictions outside this range. The technique relies on Gaussian process interpolation, and operates by relaxing the interpolation constraints outside the range of interest. This goal-oriented technique is kept simple and cheap: there are no additional parameters to set compared to the standard Gaussian process framework. The user only needs to specify a range of function values where good predictions should be obtained. The relaxation range can be selected automatically, using a scoring rule adapted to reGP models.

Such goal-oriented models can then be used in Bayesian sequential search algorithms. Here we are interested in the problem of mono-objective optimization and we propose to study the EI / EGO algorithm with such models. In a first step, we guarantee the convergence of the reGP-based algorithm on the RKHS attached to the underlying GP covariance. Then, we provide a benchmark that shows very clear benefits of using reGP models for the optimization of various functions.

## 4.7 . Properties of the truncated CRPS

We shall now write (4.33) more explicitly for the case where the range of interest is an interval $Q = (a,b)$, $-\infty \leq a < b \leq +\infty$, and provide closed-form expressions for the case where, in addition, the predictive distribution $P$ is Gaussian.

**Remark 30** *The value of the tCRPS for an interval $Q = (a,b)$ remains unchanged if the interval is closed at one or both of its endpoints.*

**Remark 31** *The value of the tCRPS for a finite (or countable) union of disjoint intervals follows readily from its values on intervals, since $Q \mapsto S_Q^{\mathrm{tCRPS}}(P,z)$ is $\sigma$-additive.*

We shall start by defining a quantity that shares similarities with (4.6).

**Definition 32** $\mathrm{EI}_q^{\uparrow}(P,z) = \mathsf{E}\left((N_1 \vee \cdots \vee N_q - z)_+\right)$ *with* $N_j \overset{\text{iid}}{\sim} P$.

The following expressions hold for a general predictive distribution $P$. With a slight abuse of notation, we will write $S_{a,b}^{\mathrm{tCRPS}}$ for $S_Q^{\mathrm{tCRPS}}$ with $Q = (a,b)$ in the following.

**Proposition 33** *Suppose that $P$ has a first order moment.*

- *Let $a,b \in \mathbb{R}$ with $a \leq b$. Then,*

$$S_{a,b}^{\mathrm{tCRPS}}(P,z) = (b \wedge z - a)_+ + \mathrm{EI}_2^{\uparrow}(P,b) - \mathrm{EI}_2^{\uparrow}(P,a) - 2\,\mathbb{1}_{z \leq b}\left(\mathrm{EI}_1^{\uparrow}(P,b) - \mathrm{EI}_1^{\uparrow}(P,a \vee z)\right). \tag{4.39}$$

- *Let $b \in \mathbb{R}$ and $N_1, N_2 \overset{\text{iid}}{\sim} P$. Then,*

$$S_{-\infty,b}^{\mathrm{tCRPS}}(P,z) = b \wedge z + \mathrm{EI}_2^{\uparrow}(P,b) - \mathsf{E}(N_1 \vee N_2) - 2\,\mathbb{1}_{z \leq b}\left(\mathrm{EI}_1^{\uparrow}(P,b) - \mathrm{EI}_1^{\uparrow}(P,z)\right). \tag{4.40}$$

- *Finally, if $a \in \mathbb{R}$, then*

$$S_{a,+\infty}^{\mathrm{tCRPS}}(P,z) = S_{-\infty,-a}^{\mathrm{tCRPS}}(\underline{P},-z), \tag{4.41}$$

*where $\underline{P}$ is the distribution of $-U$ if $U$ is $P$-distributed.*

Now, leveraging well-known analytic expressions (see, e.g., Chevalier and Ginsbourger, 2013, Nadarajah and Kotz, 2008), we have the following closed-form expressions in the Gaussian case.

**Proposition 34** *(Chevalier and Ginsbourger, 2013, Nadarajah and Kotz, 2008) Suppose that $P = \mathcal{N}(\mu,\sigma^2)$ and let $\phi$ and $\Phi$ denote respectively the pdf and the cdf of the standard Gaussian distribution. Then*

- $\mathrm{EI}_1^{\uparrow}(P,z) = \sigma\, g_1\left(\frac{\mu - z}{\sigma}\right)$, *with*

$$g_1(t) = t\,\Phi(t) + \phi(t), \tag{4.42}$$

- *for* $q \geq 2$, *we have* $\mathrm{EI}_q^{\uparrow}(P,z) = \sigma\, g_q\left(\frac{\mu - z}{\sigma}\right)$, *where*

$$g_q(t) = qt\,\Phi_q(\delta_q^t; 0, D_q D_q^{\mathsf{T}}) + q\,\Phi(-t)^{q-1}\,\phi(t) + \frac{q(q-1)}{2\sqrt{\pi}}\,\Phi_{q-1}(\delta_{q-1}^t; 0, \tfrac{1}{2}B_q), \tag{4.43}$$

*where* $\Phi_q(\cdot; m, \Sigma)$ *is the cdf of the multivariate* $\mathcal{N}(m,\Sigma)$ *distribution,*

$$B_q = 2\,\mathrm{diag}(0, \mathbb{1}_{q-2}^{\mathsf{T}}) + \mathbb{1}_{q-1}\mathbb{1}_{q-1}^{\mathsf{T}},$$

$D_q$ *is the matrix representing the linear map*

$$\mathbb{R}^q \to \mathbb{R}^q, \quad (y_1, \ldots, y_q)^{\mathsf{T}} \mapsto (-y_1, y_2 - y_1, y_3 - y_1, \ldots, y_q - y_1)^{\mathsf{T}},$$

*and* $\delta_q^t = (t, 0_{q-1}^{\mathsf{T}})^{\mathsf{T}}$,

- *finally for* $q = 2$ *we have*

$$\mathsf{E}(N_1 \vee N_2) = \mu + \frac{\sigma}{\sqrt{\pi}}. \tag{4.44}$$

For a scoring rule $S : \mathscr{P} \times \mathbb{R} \to \mathbb{R}$ and $P_1, P_2 \in \mathscr{P}$ such that $y \in \mathbb{R} \mapsto S(P_1, y)$ is $P_2$-integrable, write $S(P_1, P_2) = \mathsf{E}_{U \sim P_2}(S(P_1, U))$. The propriety of scoring rules is an important notion that formalizes "well-calibration" in the sense that a generating distribution must be identified to be optimal on average.

**Definition 35** *(see, e.g. Gneiting and Raftery, 2007) A scoring rule* $S : \mathscr{P} \times \mathbb{R} \to \mathbb{R}$ *is said to be (strictly) proper with respect to* $\mathscr{P}$ *if, for all* $P_1, P_2 \in \mathscr{P}$, *the mapping* $y \in \mathbb{R} \mapsto S(P_1, y)$ *is* $P_2$*-integrable and the mapping* $P_1 \in \mathscr{P} \mapsto S(P_1, P_2)$ *admits* $P_2$ *as a (unique) minimizer.*

A strictly proper scoring rule $S$ on a class $\mathscr{P}$ induces a *divergence*

$$(P_1, P_2) \;\mapsto\; S(P_1, P_2) - S(P_2, P_2),$$

which is non-negative on $\mathscr{P} \times \mathscr{P}$, and vanishes if and only if $P_1 = P_2$. In the case of the truncated CRPS, simple calculations lead to (Matheson and Winkler, 1976):

$$S_Q^{\mathrm{tCRPS}}(P_1, P_2) - S_Q^{\mathrm{tCRPS}}(P_2, P_2) = \int_Q \left(F_{P_1}(u) - F_{P_2}(u)\right)^2 \mathrm{d}u.$$

It follows that $S_Q^{\mathrm{tCRPS}}$ is proper for any measurable $Q \subset \mathbb{R}$, and is strictly proper with respect to the class of non-degenerate Gaussian measures on $\mathbb{R}$ as soon as $Q$ has non-empty interior.

## 4.8 . Proofs

**Lemma 36** *(Aronszajn, 1950, Section 1.5) Let $k\colon \mathbb{X}\times\mathbb{X}\to\mathbb{R}$ be a positive-definite covariance function, $\mathbb{U}\subset\mathbb{X}$, and $\mathscr{H}(\mathbb{U})$ be the RKHS attached to the restriction of $k$ to $\mathbb{U}\times\mathbb{U}$. The RKHS $\mathscr{H}(\mathbb{U})$ is the space of restrictions of functions from $\mathscr{H}(\mathbb{X})$ and the norm of $g\in\mathscr{H}(\mathbb{U})$ is given by*

$$\inf_{\tilde{g}\in\mathscr{H}(\mathbb{X}),\ \tilde{g}_{|\mathbb{U}}=g}\|\tilde{g}\|_{\mathscr{H}(\mathbb{X})}. \tag{4.45}$$

**Proof of Proposition 14**. First the existence and the uniqueness of the solution are given by the first statement of Proposition 23 (with $\mathscr{H}_{R,n}=\mathscr{H}_{R,\{x_1,\cdots,x_n\}}$).

Furthermore let $\underline{z}\in\mathbb{R}$ and write $\underline{\alpha}=K_n^{-1}\underline{z}$, the reproduction property (4.7) gives

$$\|s_{\underline{z}}\|^2_{\mathscr{H}(\mathbb{X})}=\underline{\alpha}^{\mathsf{T}}K_n\underline{\alpha}=\underline{z}^{\mathsf{T}}K_n^{-1}\underline{z}, \tag{4.46}$$

and therefore

$$\min_{h\in\mathscr{H}_{R,n}}\|h\|^2_{\mathscr{H}(\mathbb{X})}=\inf_{\underline{z}\in C_{R,n}}\ \min_{h\in\mathscr{H}(\mathbb{X}),h(\underline{x}_n)=\underline{z}}\|h\|^2_{\mathscr{H}(\mathbb{X})}=\inf_{\underline{z}\in C_{R,n}}\underline{z}^{\mathsf{T}}K_n^{-1}\underline{z},$$

where the last infimum is uniquely reached by the evaluation of the solution on $\underline{x}_n$. ■

**Proof of Proposition 15**. Write $K_{m,n}$ for the covariance matrix of the random vector $\left(\underline{Z}_m'^{\mathsf{T}},\underline{Z}_n^{\mathsf{T}}\right)^{\mathsf{T}}$. Using the equalities (4.5) and (4.46), and a slight abuse of notation by dropping irrelevant constants with respect to $\underline{z}'$ and $\underline{z}\in C_{R,n}$, we have

$$\begin{aligned}
&-2\ln\left(p\left(\underline{z}',\underline{z}\,|\,\underline{Z}_n\in C_{R,n}\right)\right)\\
&=\left(\underline{z}'^{\mathsf{T}},\underline{z}^{\mathsf{T}}\right)K_{m,n}^{-1}\left(\underline{z}'^{\mathsf{T}},\underline{z}^{\mathsf{T}}\right)^{\mathsf{T}}\\
&=\min_{h\in\mathscr{H}(\mathbb{X}),h(\underline{x}_n)=\underline{z},h(\underline{x}'_m)=\underline{z}'}\|h\|^2_{\mathscr{H}(\mathbb{X})}.
\end{aligned}$$

This gives

$$\inf_{\underline{z}'\in\mathbb{R}^m,\underline{z}\in C_{R,n}}-2\ln\left(p\left(\underline{z}',\underline{z}\,|\,\underline{Z}_n\in C_{R,n}\right)\right)=\min_{h\in\mathscr{H}(\mathbb{X}),h(\underline{x}_n)\in C_{R,n}}\|h\|^2_{\mathscr{H}(\mathbb{X})},$$

which is reached by taking $\underline{z}=\underline{z}_n^\star$ and $\underline{z}'=\left(s_{\underline{z}_n^\star}(x'_1),\ldots,s_{\underline{z}_n^\star}(x'_m)\right)^{\mathsf{T}}$. ■

**Proof of Proposition 21**. First, one has

$$\sup_{x\in\mathbb{X}}\sigma_n(x)=\sup_{x\in\mathbb{X}}\sup_{\|f\|_{\mathscr{H}(\mathbb{X})}=1}|f(x)-s_{\underline{z}_n}(x)|=\sup_{\|f\|_{\mathscr{H}(\mathbb{X})}=1}\|f-s_{\underline{z}_n}\|_{\mathbb{L}^\infty(\mathbb{X})}.$$

Now, let $f\in\mathscr{H}(\mathbb{X})$ such that $\|f\|_{\mathscr{H}(\mathbb{X})}=1$, and $\mathbb{X}^o$ be the interior of $\mathbb{X}$. The boundary of $\mathbb{X}^o$ is the one of $\mathbb{X}$ under Assumption 20, and the Sobolev space $W_2^{\alpha+d/2}(\mathbb{X}^o)$ defined by (4.18) is norm-equivalent to the Sobolev–Slobodeckij

space (see, e.g., Di Nezza et al. (2012, Proposition 3.4) for a statement on $\mathbb{R}^d$ and Grisvard (1985, Theorem 1.4.3.1) for the existence of an extension operator).

Then, one can apply (Arcangéli et al., 2007, Theorem 4.1) to $f - s_{\underline{z}_n}$ restricted to $\mathbb{X}^o$ to show that, for $h_n$ lower than some $h_0$ (not depending on $f$ or $(x_n)_{n\geq 1}$), we have:

$$\|f - s_{\underline{z}_n}\|_{\mathbb{L}^\infty(\mathbb{X})} = \|f - s_{\underline{z}_n}\|_{\mathbb{L}^\infty(\mathbb{X}^o)} \lesssim h_n^\alpha \|f - s_{\underline{z}_n}\|_{W_2^\alpha(\mathbb{X}^o)} \lesssim h_n^\alpha \|f - s_{\underline{z}_n}\|_{\mathscr{H}(\mathbb{X})} \lesssim h_n^\alpha \tag{4.47}$$

by continuity of $f - s_{\underline{z}_n}$, since $\|\cdot\|_{W_2^{\alpha+d/2}(\mathbb{X}^o)} \leq \|\cdot\|_{W_2^{\alpha+d/2}(\mathbb{X})}$ due to the definition (4.18), $W_2^{\alpha+d/2}(\mathbb{X})$ being norm equivalent to $\mathscr{H}(\mathbb{X})$, and because of the projection interpretation of $s_{\underline{z}_n}$ (see, e.g., Wendland, 2004, Theorem 13.1). Finally, one can get rid of the condition $h_n \leq h_0$ for simplicity by increasing the constant eventually, since $\sigma_n$ is bounded on $\mathbb{X}$. ■

**Proof of Proposition 23**. First observe that $\mathscr{H}_{R,\mathbb{U}}$ is not empty since it contains $f$. Furthermore, it is easy to verify that $\mathscr{H}_{R,\mathbb{U}}$ is convex and that it is closed since pointwise evaluation functionals are continuous on an RKHS. The problem is then the one of projecting the null function on a convex closed subset; hence the existence and the uniqueness.

Then, the function $s_{R,n}$ is the projection of the null function on the closed convex set $\mathscr{H}_{R,n}$ defined by (4.11). Moreover, the sequence $(\mathscr{H}_{R,n})_{n\geq 1}$ is non-increasing, so the sequence $(s_{R,n})_{n\geq 1}$ converges in $\mathscr{H}(\mathbb{X})$ to the projection of the null function on $\bigcap_{n\geq 1} \mathscr{H}_{R,n}$ (see, e.g., Brezis, 2011, Exercice 5.5), i.e. the solution of (4.23), with $\mathbb{U} = \{x_n\}$. But this last solution is also the solution on the closure since it verifies the constraints by continuity. ■

**Proof of Proposition 24**. Define $\underline{x}_n^0$ and $\underline{z}_n^0$ to be data points within $\mathbb{X}_0$, and $s_{\underline{x}_n^0,\underline{z}_n^0}$ the associated (interpolation) predictor, i.e. the solution of (4.8). Observing that $s_{\underline{x}_n^0,\underline{z}_n^0}$ interpolates $s_{R,n}$, we have:

$$\begin{aligned}
\left|f(x) - s_{R,n}(x)\right| &\leq \left|f(x) - s_{\underline{x}_n^0,\underline{z}_n^0}(x)\right| + \left|s_{\underline{x}_n^0,\underline{z}_n^0}(x) - s_{R,n}(x)\right| \\
&\leq \sigma_{n,0}(x)\,\|f\|_{\mathscr{H}(\mathbb{X}_0)} + \sigma_{n,0}(x)\,\|s_{R,n}\|_{\mathscr{H}(\mathbb{X}_0)} \\
&\leq 2\sigma_{n,0}(x)\,\|s_{R,\mathbb{X}}\|_{\mathscr{H}(\mathbb{X})},
\end{aligned}$$

since $f$ coincides with $s_{R,\mathbb{X}}$ on $\mathbb{X}_0$, $s_{R,\mathbb{X}}$ is in $\mathscr{H}_{R,n}$ (see 4.11), and by Lemma 36. ■

**Proof of Proposition 25**. For the first assertion, let $\sigma_{n,B}$ be the power function using only the observations within $B$. Using Proposition 24, the inequality $\sigma_{n,0} \leq \sigma_{n,B}$ given by the projection interpretation, and applying Proposition 21 to $B$ yields a bound depending on the fill distance $h_{n,B}$ of $\{x_1, \cdots, x_n\} \cap B$ within $B$. Finally, Lemma 37 allows us to conclude.

Regarding second assertion, $f$ is continuous so the sets $\mathbb{X}_j$ are compact for $j \geq 1$. In addition, they are disjoint so

$$\delta = \min_{1 \leq j < p} \inf_{x \in \mathbb{X}_j, y \in \mathbb{X}_p} \|x - y\| > 0.$$

Suppose now that $h_n < \delta$ and let $j \geq 1$, $x \in \mathbb{X}_j$ and $1 \leq i \leq n$ the index of the closet $x_i$ to $x$. By definition, $\|x - x_i\| \leq h_n$ and therefore $x_i \in \mathbb{X}_0 \cup \mathbb{X}_j$. Let $\mathscr{H}^\star(\mathbb{X})$ be the (topological) dual of $\mathscr{H}(\mathbb{X})$ and

$$\delta_y \colon g \in \mathscr{H}(\mathbb{X}) \mapsto g(y), \tag{4.48}$$

which lies in $\mathscr{H}^\star(\mathbb{X})$ for all $y \in \mathbb{X}$. Then using the reproducing property (4.7), we have

$$|s_{R,n}(x_i) - s_{R,n}(x)| \leq \|\delta_{x_i} - \delta_x\|_{\mathscr{H}^\star(\mathbb{X})} \|s_{R,n}\|_{\mathscr{H}(\mathbb{X})} \leq \|\delta_{x_i} - \delta_x\|_{\mathscr{H}^\star(\mathbb{X})} \|s_{R,\mathbb{X}}\|_{\mathscr{H}(\mathbb{X})},$$

and therefore

$$d(s_{R,n}(x), R_j) \leq \|\delta_{x_i} - \delta_x\|_{\mathscr{H}^\star(\mathbb{X})} \|s_{R,\mathbb{X}}\|_{\mathscr{H}(\mathbb{X})} + d(s_{R,n}(x_i), R_j).$$

Now, if $x_i \in \mathbb{X}_j$, then $d(s_{R,n}(x_i), R_j) = 0$. Otherwise, $x_i \in \mathbb{X}_0$ necessarily and then, using the fact that $s_{R,\mathbb{X}}(x) \in R_j$, we have:

$$\begin{aligned} d(s_{R,n}(x_i), R_j) &\leq |s_{R,n}(x_i) - s_{R,\mathbb{X}}(x)| \\ &= |s_{R,\mathbb{X}}(x_i) - s_{R,\mathbb{X}}(x)| \\ &\leq \|\delta_{x_i} - \delta_x\|_{\mathscr{H}^\star(\mathbb{X})} \|s_{R,\mathbb{X}}\|_{\mathscr{H}(\mathbb{X})}. \end{aligned}$$

So one can use Lemma 38 along with the previous statements to conclude if $h_n < \delta$.

Finally, treating the case $h_n \geq \delta$ is straightforward using the reproducing property (4.7), the fact that $\sup_{x \in \mathbb{X}} \sqrt{k(x,x)}$ is finite thanks to the compacity of $\mathbb{X}$, and $d(s_{R,n}(x), R_j) \leq |s_{R,n}(x) - s_{R,\mathbb{X}}(x)|$ for $j \geq 1$ and $x \in \mathbb{X}_j$. ■

**Lemma 37** *Let $B \subset \mathbb{X}$ verifying Assumption 20 and $h_{n,B}$ be the fill distance of $\mathbb{X}_{n,B} = \{x_1, \cdots, x_n\} \cap B$ within $B$, with the convention $h_{n,B} = \operatorname{diam}(B)$ if $\mathbb{X}_{n,B}$ is empty. Then, $h_{n,B} \lesssim h_n$.*

**Proof**. The idea of the proof is given by Wendland (2004, Lemma 11.31), but it is interlinked with a much more sophisticated construction, so we provide a specific version here for completeness. Adams and Fournier (2003, Section 4.11) shows that $B$ verifies a cone condition with radius $\rho > 0$ and angle $\phi \in (0, \pi/2)$. If $\mathbb{X}_{n,B}$ is not empty, then the compacity of $B$ ensures the existence of an $x \in B$ such that $d(x, \mathbb{X}_{n,B}) = h_{n,B}$. (If $\mathbb{X}_{n,B}$ is empty, then the rest of the proof is also valid taking an arbitrary $x \in B$.)

A cone $C$ originating from $x$ with angle $\phi$ and radius $\delta = \min(h_{n,B}, \rho)$ is contained in $B$ and its interior do not contains observations. Furthermore, Wendland (2004, Lemma 3.7) shows that there exists a $y \in C$ such that the open ball $B\left(y, \delta\sin(\phi)\left(1+\sin(\phi)\right)^{-1}\right)$ is subset of $C$, and therefore contains no observations as well. Thus, $h_n \geq \delta\sin(\phi)(1+\sin(\phi))^{-1}$. Now, if $h_{n,B} \leq \rho$, then the desired result follows. If not, the result holds as well since $h_{n,B} \leq \operatorname{diam}(B)$. ■

**Lemma 38** *If $k$ has smoothness $\alpha > 0$, then there exists $h_0 > 0$ depending only on $\alpha$ such that, for all $x, y \in \mathbb{X}$ verifying $\|x-y\| \leq h_0$, we have*

$$\|\delta_y - \delta_x\|_{\mathscr{H}^\star(\mathbb{X})} \lesssim \|x-y\|^\alpha, \text{ for } \alpha < 1, \tag{4.49}$$

$$\|\delta_y - \delta_x\|_{\mathscr{H}^\star(\mathbb{X})} \lesssim \sqrt{|\ln(\|x-y\|)|}\,\|x-y\|, \text{ for } \alpha = 1, \tag{4.50}$$

*and*

$$\|\delta_y - \delta_x\|_{\mathscr{H}^\star(\mathbb{X})} \lesssim \|x-y\|, \text{ for } \alpha > 1. \tag{4.51}$$

**Proof**. Since equivalent norms give equivalent operator norms on the topological dual of a normed space, it suffices to show the result for a unit-variance isotropic Matérn covariance function (4.4) of regularity $\alpha$.

In this case, we have

$$\|\delta_y - \delta_x\|^2_{\mathscr{H}^\star(\mathbb{X})} = k(x,x) + k(y,y) - 2k(x,y) = 2\left(1 - r_\alpha(\|x-y\|)\right),$$

with $r_\alpha$ the corresponding isotropic correlation function. Standard results on principal irregular terms (see, e.g., Stein, 1999, Chapter 2.7) give the results. ■

**Lemma 39** *If $g_1, g_1 \in W_2^\gamma(\mathbb{X})$ for $\gamma > d/2$, then $g_1 g_2 \in W_2^\gamma(\mathbb{X})$.*

**Proof**. By the definition (4.18) of $W_2^\gamma(\mathbb{X})$, the functions $g_1$ and $g_2$ can be extended as functions on $\mathbb{R}^d$, having their product in $W_2^\gamma(\mathbb{R}^d)$ (Strichartz, 1967, Theorem 2.1). Taking the restriction shows the desired result. ■

**Proof of Proposition 26**. We use a bump function argument. Let $B(x_0, r) \subset \mathbb{X}_j$ (with $r > 0$) be an open ball. There exists a $C^\infty$ function $\phi : \mathbb{R}^d \to \mathbb{R}$ such that

$$\begin{cases} 0 \leq \phi \leq 1, & \\ \phi(x) = 1 & \text{only if } \quad x = 0, \\ \phi(x) = 0 & \text{if } \qquad x \in \mathbb{X} \setminus B(0,r). \end{cases}$$

Let $c \in R_j \setminus \{f(x_0)\}$, $\phi_n = \phi\left(n(\cdot - x_0)\right)$ as a function on $\mathbb{X}$, and $f_n = (1-\phi_n) f + c\phi_n$, for $n \geq 1$. We have $\phi_n \in W_2^{\alpha+d/2}(\mathbb{R}^d)$ as a function on $\mathbb{R}^d$, so it belongs to $W_2^{\alpha+d/2}(\mathbb{X})$ as a function on $\mathbb{X}$, and Lemma 39 ensures that $f_n \in \mathscr{H}(\mathbb{X})$.

Moreover, it is easy to check that $f_n \in \mathscr{H}_{R,\mathbb{X}}$. Observe that the sequence $(f_n)_{n\geq 1}$ converges pointwise to a discontinuous function that lies thus outside $\mathscr{H}(\mathbb{X})$.

Suppose now that $\|f_n\|_{\mathscr{H}(\mathbb{X})} \nrightarrow +\infty$ then one can extract a bounded subsequence of norms and a classical weak compacity argument would yield a weakly convergent subsequence, which is impossible since the pointwise limit is not in $\mathscr{H}(\mathbb{X})$. ■

**Proof of Proposition 29**. This is an adaptation of Theorem 6 from (Vazquez and Bect, 2010b). For $f \in \mathscr{H}(\mathbb{X})$, write $(x_n)_{n\geq 1}$ for the corresponding sequence $(X_n)_{n\geq 1}$ generated by EGO-R. Moreover, write $s_n = s_{R_n,n}$ for the reGP predictor at the step $n$ to avoid cumbersome notations. Then, for $x \in \mathbb{X}$, write $\rho_{n,t_n}(x) = \gamma(m_n - s_n(x), \sigma_n^2(x))$ for the expected improvement under the reGP predictive distribution, with $\gamma$ the function defined in Proposition 12.

Suppose that there exists some $x_0 \in \mathbb{X}$ such that $\sigma_n^2(x_0) \geq C_1 > 0$. The sequence $(m_n)_{n\geq 1}$ converges and the reproducing property (4.7) yields

$$|s_n(x_0)| \leq \sqrt{k(x_0,\, x_0)}\|s_n\|_{\mathscr{H}(\mathbb{X})} \leq \sqrt{k(x_0,\, x_0)}\|f\|_{\mathscr{H}(\mathbb{X})},$$

so the sequence $(|m_n - s_n(x_0)|)_{n\geq 1}$ is bounded by, say $C_2$. We have then

$$v_n = \sup_{x\in\mathbb{X}} \rho_{n,t_n}(x) \geq \gamma\big(m_n - s_n(x_0), \sigma_n^2(x_0)\big) \geq \gamma(-C_2, C_1) > 0$$

by Proposition 12. But this yields a contradiction with Lemma 41, so the decreasing sequence $\left(\sigma_n^2\right)_{n\geq 1}$ converges pointwise on $\mathbb{X}$ to zero. Proposition 10 from Vazquez and Bect (2010b) then implies that every $x \in \mathbb{X}$ is adherent to $\{x_n\}$. ■

**Lemma 40** *Use the notations of Proposition 29 and let $(y_n)_{n\geq 1}$ be a sequence in $\mathbb{X}$. Assume that the sequence $(y_n)_{n\geq 1}$ is convergent, denote by $y^\star$ its limit and assume that $y^\star$ is an adherent point of the set $\{x_n\}$. Let $t_\infty = \liminf t_n$, then*

- $s_n(y_n) \to f(y^\star)$ *if* $f(y^\star) < t_\infty$,
- $\liminf s_n(y_n) \geq t_\infty$, *otherwise.*

*In particular, we have*

$$\liminf s_n(y_n) \geq \min(f(y^\star),\, t_\infty). \tag{4.52}$$

**Proof**. If $y^\star \in \{x_n\}$, then the result holds so suppose the converse. Let $x_{\phi(n)}$ be a subsequence converging to $y^\star$ and let

$$\psi(n) = \max\{\phi(k), \phi(k) \leq n\}.$$

We proceed as in Proposition 25. First,

$$\big|s_n(x_{\psi(n)}) - s_n(y_n)\big| \leq \|\delta_{x_{\psi(n)}} - \delta_{y_n}\|_{\mathscr{H}^\star(\mathbb{X})}\|s_n\|_{\mathscr{H}(\mathbb{X})} \leq \|\delta_{x_{\psi(n)}} - \delta_{y_n}\|_{\mathscr{H}^\star(\mathbb{X})}\|f\|_{\mathscr{H}(\mathbb{X})}$$

converges to zero thanks to Lemma 38.

Now $s_n(x_{\psi(n)}) \geq \min\left(f\left(x_{\psi(n)}\right), t_n\right)$ so $\liminf s_n(x_{\psi(n)}) \geq \min\left(f\left(y^\star\right), t_\infty\right)$, which gives the second assertion. Observe that $f(x_{\psi(n)}) < t_n$ ultimately if $f(y^\star) < t_\infty$ for the first assertion. ■

**Lemma 41** *Using the notations of Proposition 29 and writing* $v_n = \sup_{x\in\mathbb{X}} \rho_{n,t_n}(x)$, *we have* $\liminf v_n = 0$.

**Proof**. This is an adaptation of Lemma 12 from (Vazquez and Bect, 2010b).

Let $y^\star$ be a cluster point of $(x_n)_{n\geq 1}$ and let $x_{\phi(n)}$ be a subsequence converging to $y^\star$. We are going to prove that $v_{\phi(n)-1} \to 0$. Define

$$z_{\phi(n)-1} = m_{\phi(n)-1} - s_{\phi(n)-1}(x_{\phi(n)}).$$

Lemma 40 gives $\liminf s_{\phi(n)-1}(x_{\phi(n)}) \geq \min(f(y^\star),\ t_\infty)$, with $t_\infty = \liminf t_n$. Moreover we have

$$m_{\phi(n)-1} \leq \min\left(f\left(x_{\phi(n-1)}\right),\ t_{\phi(n)-1}\right)$$

so $\lim m_{\phi(n)-1} \leq \min(f(y^\star),\ t_\infty)$ since $\left(m_{\phi(n)-1}\right)_{n\geq 1}$ is non-increasing. The previous arguments show that $\limsup z_{\phi(n)-1} \leq 0$. The latter fact and $\sigma^2_{\phi(n)-1}(x_{\phi(n)}) \to 0$ (Vazquez and Bect, 2010b, Proposition 10) show that

$$v_{\phi(n)-1} = \gamma\left(z_{\phi(n)-1},\ \sigma^2_{\phi(n)-1}(x_{\phi(n)})\right) \leq \gamma\left(\sup_{k\geq n} z_{\phi(k)-1},\ \sigma^2_{\phi(n)-1}(x_{\phi(n)})\right) \to 0,$$

since $\gamma$ is non-decreasing with respect to its first argument and continuous. ■

**Lemma 42** *Assume that* $b$ *is finite, and that either* $a$ *is finite too or* $\int_{-\infty}^0 F_P(u)\,\mathrm{d}u = \int_{-\infty}^0 |u|\,P(\mathrm{d}u)$ *is finite. Then*

$$S^{\mathrm{tCRPS}}_{a,b}(P,z) \;=\; (b - a\vee z)_+ + R_2(a,b) - 2R_1(a\vee z, b), \tag{4.53}$$

*where*

$$R_q(a,b) \;=\; \int_{-\infty}^{+\infty} \mathbb{1}_{a\leq u\leq b}\, F_P(u)^q\,\mathrm{d}u. \tag{4.54}$$

**Proof**.

$$\begin{aligned}
S^{\mathrm{tCRPS}}_{a,b}(P,z) \;&=\; \int_{-\infty}^{+\infty} \mathbb{1}_{a\leq u\leq b}\,\left(\mathbb{1}_{z\leq u} - F_P(u)\right)^2\,\mathrm{d}u \\
&=\; \int_{-\infty}^{+\infty} \mathbb{1}_{a\leq u\leq b}\,\left(\mathbb{1}_{z\leq u} + F_P(u)^2 - 2\,\mathbb{1}_{z\leq u}\,F_P(u)\right)\,\mathrm{d}u \\
&=\; \int_{-\infty}^{+\infty} \left(\mathbb{1}_{a\vee z\leq u\leq b} + \mathbb{1}_{a\leq u\leq b}\,F_P(u)^2 - 2\,\mathbb{1}_{a\vee z\leq u\leq b}\,F_P(u)\right)\,\mathrm{d}u \\
&=\; (b - a\vee z)_+ + R_2(a,b) - 2R_1(a\vee z, b).
\end{aligned}$$

■

**Lemma 43** *Let $a,b \in \mathbb{R}$ with $a \leq b$. Let $q \geq 1$. Then*

$$R_q(a,b) \;=\; b-a+\mathrm{EI}_q^{\uparrow}(P,b)-\mathrm{EI}_q^{\uparrow}(P,a). \tag{4.55}$$

**Proof**.

$$\begin{aligned}
R_q(a,b) \;&=\; \int \mathbb{1}_{a\leq u\leq b}\, F_P(u)^q \,\mathrm{d}u \\
&=\; \int \mathbb{1}_{a\leq u\leq b} \prod_{j=1}^{q} \mathsf{E}\left(\mathbb{1}_{N_j\leq u}\right) \mathrm{d}u \qquad \text{with } N_j \overset{\text{iid}}{\sim} P \\
&=\; \mathsf{E}\left(\int \mathbb{1}_{a\vee N_1\vee\cdots\vee N_q\leq u\leq b}\,\mathrm{d}u\right) \\
&=\; \mathsf{E}\left((b-a\vee N_1\vee\cdots\vee N_q)_+\right) \\
&=\; b-a+\mathrm{EI}_q^{\uparrow}(P,b)-\mathrm{EI}_q^{\uparrow}(P,a).
\end{aligned}$$

. ■

**Proof of Proposition 33**. The first result is given by Lemma 42 and Lemma 43.

Then using the dominated convergence theorem, it is easy to see that, when $a \to -\infty$

$$\mathrm{EI}_q^{\uparrow}(P,a) \;=\; \mathsf{E}\left(N_1\vee\cdots\vee N_q\right)-a+o(1), \tag{4.56}$$

and therefore

$$R_q(-\infty,b) \;=\; \lim_{a\to-\infty} R_q(a,b) \;=\; b+\mathrm{EI}_q^{\uparrow}(P,b)-\mathsf{E}\left(N_1\vee\cdots\vee N_q\right), \tag{4.57}$$

which gives the second statement.

Finally, a change of variable gives

$$\int_a^{+\infty} (F_P(u)-\mathbb{1}_{z\leq u})^2\,\mathrm{d}u = \int_{-\infty}^{-a} (F_{\underline{P}}(u)-P(U=-u)-\mathbb{1}_{-z<u})^2\,\mathrm{d}u,$$

and the last statement follows by observing that a probability measure admits at most a countable number of atoms. ■

## 4.9 . Optimization benchmark results

The results are provided in Figure 4.10, Figure 4.11, Figure 4.12, and Figure 4.13, for the other test functions from Table 4.1, using the same format as in Figure 4.8. Observe that the two heuristics for reGP yield (sometimes very) substantial improvements on Zakharov $(4)$, Zakharov $(10)$, Three-hump Camel, Perm $(4)$, Perm $(6)$, Rosenbrock $(4)$, and Rosenbrock $(6)$. However, only the "Concentration" heuristic yields clear benefits for Zakharov $(6)$, Dixon-Price $(4)$, Dixon-Price $(6)$, and Rosenbrock $(10)$. Furthermore, the "Concentration" heuristic underperforms slightly on the (multimodal) Shekel problems. Finally, the EGO and EGO-R algorithms yield indistinguishable results on the remainings cases.

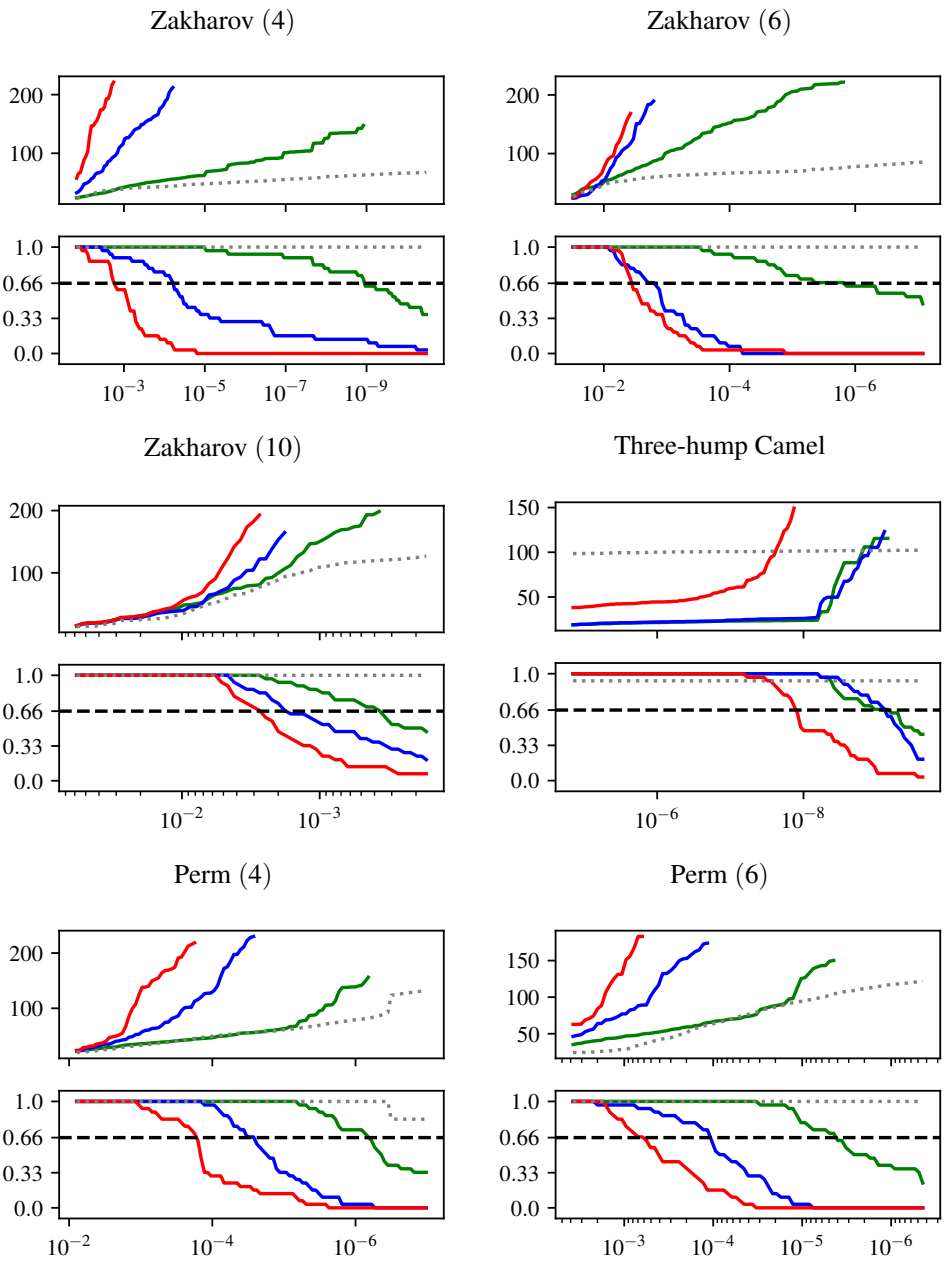



Figure 4.10: Same as Figure 4.8. The gray dashed line stands for the Dual simulated Annealing algorithm, the red line for the standard EGO algorithm, and the blue and green lines for EGO-R, with the “Constant” and “Concentration” heuristics.

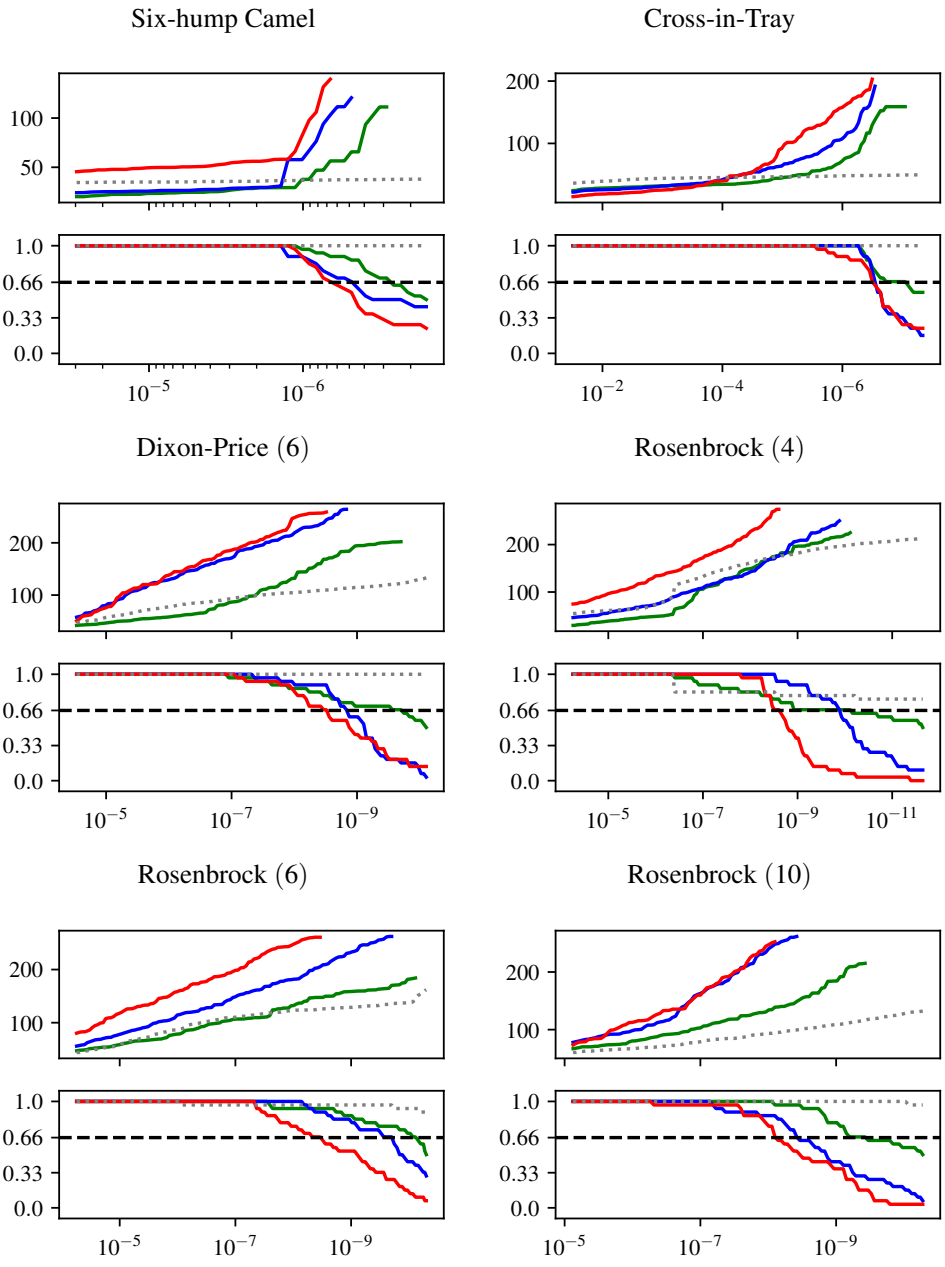



Figure 4.11: Same as Figure 4.8. The gray dashed line stands for the Dual simulated Annealing algorithm, the red line for the standard EGO algorithm, and the blue and green lines for EGO-R, with the "Constant" and "Concentration" heuristics.

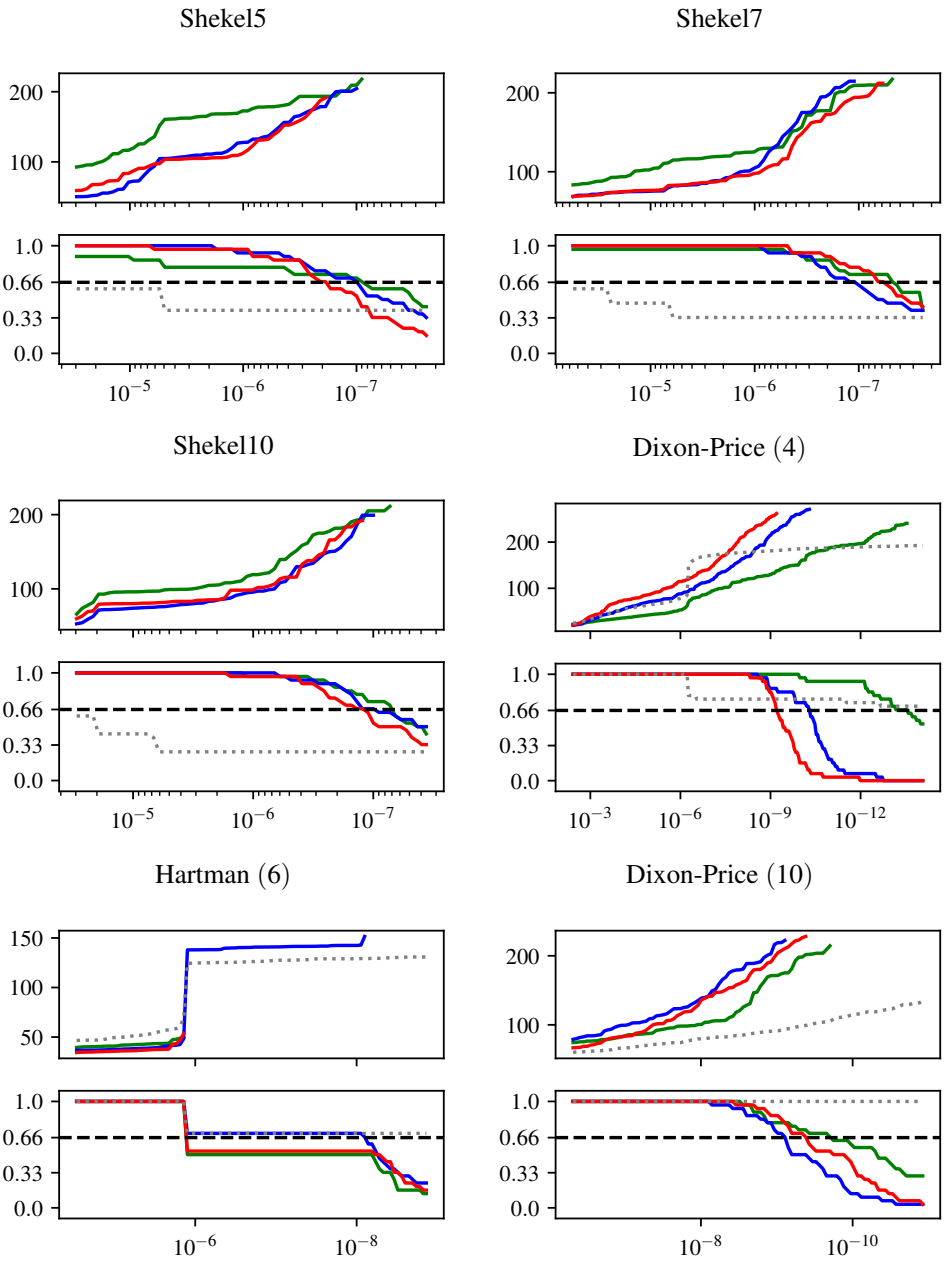



Figure 4.12: Same as Figure 4.8. The gray dashed line stands for the Dual simulated Annealing algorithm, the red line for the standard EGO algorithm, and the blue and green lines for EGO-R, with the “Constant” and “Concentration” heuristics.

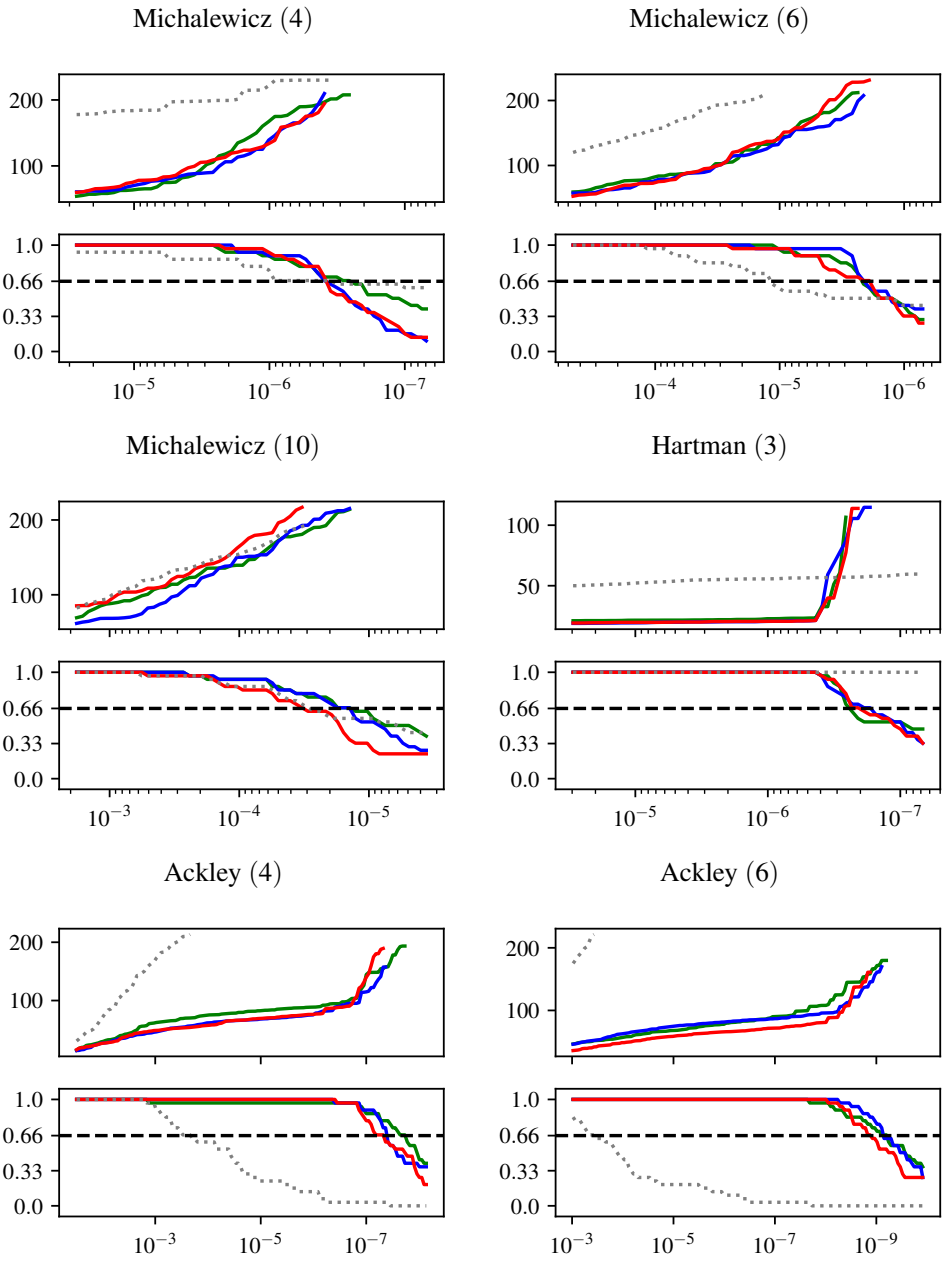



Figure 4.13: Same as Figure 4.8. The gray dashed line stands for the Dual simulated Annealing algorithm, the red line for the standard EGO algorithm, and the blue and green lines for EGO-R, with the “Constant” and “Concentration” heuristics.

# Part IV

# Conclusions and perspectives

## 1 . Contributions

This thesis focuses on the choice of a Gaussian model for predicting an unknown function. More specifically, the work goes in two directions: 1) questioning the standard methods for model selection; and 2) contributing to the field of non-stationary modeling by considering goal-oriented approaches.

The first task aims at improving model selection in applications. Starting with parameter selection by maximum likelihood estimation, inconsistencies across different softwares seemed to be the first issue to address. Chapter 2 traces the origin of these inconsistencies to the numerical noise amplified by the condition number of the covariance matrix. Then, simple levers are proposed to mitigate this issue and lead to clear benefits. Although this part of the manuscript deals with a rather prosaic issue, it contains ingredients that are fundamental for building efficient algorithms relying on Gaussian processes. For instance, potentially poor results can be expected when using a Bayesian optimization algorithm with the default settings of Python packages mentioned in Chapter 2. Using a robust implementation is a key requirement to benefit from the methodologies proposed in this manuscript.

A popular alternative to maximum likelihood estimation for selecting the parameters of a stationary Gaussian process is leave-one-out cross-validation, introduced for computer experiments by Currin et al. (1988), thanks to the existence of fast formulas that can be traced back at least to Dubrule (1983). The second contribution of this manuscript concerns the efficient computation of gradients of cross-validation criteria. Rasmussen and Williams (2006) deemed that the computation of the gradients of cross-validation criteria are unavoidably more expensive than that of the likelihood. Chapter 3 shows that this is not true. The chapter also provides a probabilistic interpretation of the fast cross-validation formulas, using the notion of precision matrices (see, e.g., Rue and Held, 2005, Section 2.2). Eventually, a toy example at the end of the chapter suggests that $K$-fold cross-validation may sometimes help, as previously discussed by Ginsbourger and Schärer (2021).

Overall, Chapter 2 and Chapter 3 make computational contributions to the topic of parameter selection. In Chapter 1, we compare the relative performances of standard and new selection criteria. Intensive numerical experiments were conducted to investigate this issue. Another question emerged in this study about the influence of the choice of the family of covariance functions. More specifically, analyzing the performances of stationary models on test data according to various scoring rules led to the conclusion that the choice of a selection criterion has less influence than the regularity parameter of the Matérn covariance function, especially if the function to be inferred is smooth. Furthermore, the numerical results show that the regularity parameter can be successfully chosen by the criteria with almost no loss of performance compared to the ideal situation where one knows in advance which value to use. Recent theoretical results from Kar-

vonen (2022) support this observation. Moreover, in our experiments, maximum likelihood estimation stands out as a good selection criterion. Another interesting selection criterion is the LOO-CRPS. Interestingly, automatically selecting the regularity in the constrained mono-objective optimization benchmark used by Feliot et al. (2017) showed significant benefits. Unfortunately, the regularity parameter controls the model globally. The rough behavior of a function may be localized in a region, and using a smoother model exclusively outside this region could be very beneficial compared to using a globally rough model.

This observation suggests that some way of localizing the models could be very beneficial. Numerous approaches are referenced in the literature review from Chapter 4. Overall, this review shows that building non-stationary models must achieve a trade-off between the expressiveness of the model and the dimension of the parameter space. However, to the best of our knowledge, very few authors explored the path of goal-oriented modeling. Chapter 4 proposes a new goal-oriented approach for modeling unknown functions, which turns out to be very beneficial for Bayesian optimization. The method, called reGP, is built to improve the fit in an (output) range of interest by transforming the data lying outside this range. It can be seen as an instance of the two categories of approaches mentioned in Section 4.1.2. First, it is a local approach since it implicitly involves an input region of interest (but this region can be very complex, especially when the dimension is high). Second, it involves a transformation of the output, but which does not act on the range where the user wants improved predictions.

The transformation acts on a range called the *relaxation range*. It may be interpreted in two ways: 1) as a relaxation of the problem of minimal-norm interpolation in the RKHS; and 2) as an approximation of a constrained Gaussian process model (see, e.g., Da Veiga and Marrel, 2012). This is implemented by extending the likelihood optimization formulation to include "relaxed" observations that must be optimized as well. The method also maintains a reasonable complexity that makes it possible to propose an adaptive approach for the choice of the relaxation set, thanks to a scoring rule tailored to the problem of goal-oriented modeling. Furthermore, combining the two previous contributions with the formalism of expected improvement (Mockus et al., 1978) yields an optimization algorithm called EGO-R, which outperforms EGO on a benchmark. The interest of reGP is further illustrated by looking at the transformation learned, which resembles some common transformations traditionally used in a few examples.

The benefits of reGP are also illustrated by theoretical analysis. A first result shows the convergence of EGO-R when the function lies in the RKHS attached to a zero-mean Gaussian process with a fixed covariance function. Then, error bounds show why improved fit in the range of interest is obtained.

Although the reGP predictive distribution is a Gaussian process, it does not have (or at least, we could not find) a proper Bayesian interpretation. Nevertheless, it can be seen as a probabilistic modeling tool within the formalism described in

Section 2.1. While being applied mainly to mono-objective Bayesian optimization in this work, it can be applied to the estimation of excursion sets and, of course, more generally for any application in which the practitioner can set an explicit range of interest.

### 2 . A focus on numerical simulators

Let us now discuss the implications for modeling numerical simulators. First, the numerical ingredients presented in Chapter 2 are a step towards allowing non-statistician users to use GPs in a robust way. Then, automatically choosing the regularity value is another step in this direction. For instance, choosing a high regularity value makes it possible to model well a smooth numerical simulator with a smaller number of observations. Conversely, choosing a small regularity value makes it possible to mitigate the difficulties posed by a numerical simulator exhibiting a discontinuity. Such discontinuities are frequent and can be caused by the physical model itself or by a numerical PDE solver. If the discontinuity lies in a non-interesting (output) range of values, then the reGP method allows to obtain even better performances. For example, one type of practical cases encountered by Safran consists in numerical codes that no longer make physical sense for some combinations of input values that, e.g., yield too high values for an output. This set of input combinations can be difficult to characterize and the reGP method allows to circumvent this issue.

### 3 . Limitations and future works

Several promising tracks for future research could be explored in the future.

First of all, although the numerical recipes from Chapter 2 implemented within (a custom version of) the package GPy (Sheffield machine learning group, 2012–2020) are effective, the underlying issue of the noisy likelihood could be addressed in greater depth. We believe that this topic is of primary concern and that there is a large room for improvement in optimizing the likelihood. Furthermore, our numerical experiments suggest that the numerical issues are caused by maximum likelihood estimates occurring for very large correlation lengths, which brings the model closer to a spline (see, e.g., Barthelmé et al., 2022, Kimeldorf and Wahba, 1970). This issue has recently started to be investigated by Karvonen and Oates (2022).

Chapter 3 presents a computational contribution for selecting the parameters of a Gaussian process with cross-validation criteria. Unfortunately, the leave-one-out cross-validation criteria turn out to be of limited interest in our numerical experiments. Although other cross-validation schemes were not tested, lower accuracy bounds suggest that there is little room for improvement with respect to maximum likelihood. However, the numerical experiment from Chapter 3 shows an example

where a $K$-fold cross-validation criterion improves long-distance predictions when the data is clustered. In this spirit, it could be possible to build cross-validation criteria dedicated to optimization by working on the cross-validation scheme rather than on the loss function as done in Part III. For example, an interesting idea would be to validate increments, in the spirit of Picheny et al. (2019).

The practical recommendation emerging from the simulations presented in Chapter 1 has a computational downside: the regularity parameter is selected using a double-loop on top of the standard continuous optimization for the variance and the correlation lengths. Therefore, the computational time is multiplied by a number of candidates values for the regularity. Although the results suggest that sticking to the values $\{1/2, 3/2, 5/2, 7/2, \infty\}$ provides satisfactory results, breaking this loop could yield computational savings and make it possible to explore other values. However, no closed-form expression exists for the Matérn covariance function and its gradient when the regularity parameter is not a half-integer. Nevertheless, some packages (see, e.g., STK, Bect et al., 2011–2021) allow to estimate $\nu$ continuously thanks to finite-difference numerical approximations (see also Geoga et al., 2022).

Moreover, the numerical experiments described in Chapter 1 could be enriched by a comparison with regularization (see, e.g., Lizotte et al., 2011) or fully Bayesian (see, e.g., Benassi et al., 2011, Handcock and Stein, 1993) methods, that could be especially interesting when the number of observations is low. In addition, considering non-uniform designs and/or noisy observations could provide more insights. Theoretical asymptotic results could also provide additional support to the findings. Several steps were made in this direction recently by Karvonen (2022), Karvonen et al. (2020), and other works have been done in the case of noisy observations (see, e.g., van Der Vaart and van Zanten, 2011, van der Vaart and van Zanten, 2009, Wynne et al., 2021).

Regarding Part III, the main perspective is to extend reGP to the case of noisy observations. One of the main difficulties comes from the fact that taking noise into account already consists in relaxing the interpolation constraints with a quadratic data fitting term. However, the relaxation due to Gaussian noise is symmetric and involves the exact values of the data. We must therefore find a way to conciliate these two relaxations.

Moreover, even in the noiseless case, the reGP methodology can be improved. Indeed, the choice of the validation threshold $t_n^{(0)}$ was only briefly discussed in Chapter 4, but we can see that it has a significant impact on the benchmark results. More precisely, the heuristic called “Constant” never under-performs significantly compared to EGO and sometimes gives very significant improvements. The "Concentration" heuristic, on the other hand, sometimes gives more spectacular results but under-performs slightly for the optimization of functions with many local minima. More advanced heuristics can probably be found with ideas from the field of trust-region-based algorithms (Conn et al., 2000). Eventually, we believe that a

theoretical study would be beneficial.

The procedure for automating the selection of the relaxation range can also be improved since it implies building several model candidates by looping over a set of threshold candidates $t_n^{(0)} < \cdots < t_n^{(G)}$. (Note that selecting $\nu$ as in Chapter 1 becomes then impractical, although it may be interesting in principle to select also this parameter when using reGP.) Offering the possibility to choose the relaxation threshold jointly with the process parameters would allow a significant gain in computation time. Nevertheless, the non-smooth effect of $t_n$ on the predictions makes gradient-based approaches difficult to implement. Studying empirically the impact of the number $G$ of threshold candidates on the results could be a simpler solution.

It would also be appreciable to extend the use of reGP to constrained or multi-objective optimization problems. For the constrained case, it seems natural to use the approach described in Section 4.4.4. Nevertheless, it could be necessary, in some cases, to consider the functions of the problem simultaneously by defining relaxations that apply jointly to the objectives and the constraints. Generalizing the method will thus require finding heuristics for defining multivariate regions of interest $Q_n \subset \mathbb{R}^q$ and sequences of candidate relaxation regions

$$\mathbb{R}^q \setminus Q_n = R_n^{(0)} \supset R_n^{(1)} \supset \cdots \supset R_n^{(G)} = \emptyset$$

from which to select a region $R_n^{(g)}$ to be accurate on $Q_n$.

# A - Synthèse

Cette thèse porte sur la modélisation par processus gaussien de simulateurs numériques utilisés pour la conception industrielle. Le temps de calcul de ces simulateurs pouvant atteindre jusqu'à plusieurs heures, l'utilisation de processus gaussiens s'avère souvent efficace pour mener à bien une tâche de conception à partir d'un petit nombre d'exécutions du simulateur. Plus précisément, ce manuscrit étudie le problème du choix et de la validation de modèle et s'articule autour de deux axes.

Dans un premier axe, nous étudions de manière empirique l'impact de plusieurs facteurs sur la prédiction par processus gaussien stationnaire. Un processus gaussien stationnaire est souvent choisi au sein d'une famille paramétrée en optimisant un critère de sélection. Nous étudions principalement deux facteurs dans cette partie : le choix du critère de sélection (vraisemblance, validation croisée...) et le choix d'une famille de processus. Pour le dernier facteur, cette partie se concentre sur le choix du paramètre de régularité d'une fonction de covariance de Matérn (1986). Des expériences numériques sur plusieurs cas issus de la littérature des «computer experiments» sont menées et fournissent plusieurs conclusions. La première est que le choix du paramètre de régularité a un impact plus important que le choix d'un critère de vraisemblance ou de validation croisée. Ensuite, les résultats montrent que le paramètre de régularité peut-être sélectionné de manière satisfaisante par les critères. Enfin, une analyse au second ordre montre que le critère de vraisemblance donne des précisions similaires voire supérieures à celles des autres critères. Ce travail de simulation numérique a nécessité des contributions annexes, également présentées dans cette partie. Tout d'abord, le Chapitre 2 présente les difficultés numériques importantes posées par le problème d'optimisation de la vraisemblance à travers des expériences numériques montrant que ces difficultés sont liées au bruit numérique dont l'amplitude peut-être reliée au conditionnement de la matrice de covariance. Ensuite, le Chapitre 3 revisite les formules analytiques efficaces pour le calcul de critères de validation croisée et propose des formules analogues, plus efficaces que celles référencées dans la littérature, pour le calcul de gradients.

Une approche *ciblée* de modélisation par processus gaussien est proposée dans la deuxième partie de ce manuscrit. De manière générale, cette approche permet d'obtenir des modèles plus prédictifs sur une plage d'intérêt en sortie. L'idée principale est de relâcher les contraintes d'interpolation sur une plage de valeur disjointe de la plage d'intérêt. Une procédure de validation croisée est proposée pour le choix automatique de la plage de relaxation. Des simulations numériques montrent l'intérêt de cette méthode, en conjonction avec le critère «Expected Improvement», pour l'optimisation Bayésienne lorsque l'on cible, par exemple, les valeurs basses dans le cadre d'une minimisation. La convergence de l'algorithme est établie lorsque la fonction appartient à l'espace de Hilbert à noyau reproduisant

associé à la fonction de covariance. Des bornes d'erreur montrant le gain en précision sur la plage d'intérêt sont également fournies.

Les contributions évoquées présentent plusieurs limitations. Premièrement, les expériences numériques sur la modélisation stationnaire ne traitent ni le cas d'un bruit d'observation, ni les approches complètement bayésiennes. De plus, le choix du paramètre de régularité est opéré à l'aide d'une «boucle» sur un ensemble fini, dont la taille fait augmenter linéairement le temps de calcul; une approche conjointe permettant de sélectionner ce paramètre continûment serait appréciable. En outre, la méthode ciblée proposée présente également quelques limitations. Premièrement, cette technique ne s'applique pas au cas bruité dans lequel les contraintes d'interpolation sont déjà relâchées. Par ailleurs, son utilisation pour l'optimisation bayésienne nécessite de choisir un seuil de validation dynamiquement, et les heuristiques proposées restent probablement perfectibles. Enfin, le choix automatique de la plage de relaxation implique également une boucle sur un ensemble de candidats dont la taille démultiplie le temps de calcul. Des résultats numériques satisfaisants sont obtenus en se restreignant à une dizaine de candidats.

# Bibliography


R. A. Adams and J. J. F. Fournier. *Sobolev spaces*. Elsevier, 2003.

D. M. Allen. *The prediction sum of squares as a criterion for selecting predictor variables*. University of Kentucky, 1971.

D. M. Allen. The relationship between variable selection and data agumentation and a method for prediction. *Technometrics*, 16(1):125–127, 1974.

I. Andrianakis and P. G. Challenor. The effect of the nugget on Gaussian process emulators of computer models. *Computational Statistics and Data Analysis*, 56 (12):4215 – 4228, 2012.

R. Arcangéli, M. C. López de Silanes, and J. J. Torrens. An extension of a bound for functions in Sobolev spaces, with applications to (m, s)-spline interpolation and smoothing. *Numerische Mathematik*, 107(2):181–211, 2007.

S. Arlot and A. Celisse. A survey of cross validation procedures for model selection. *Statistics Surveys*, 4, 07 2009.

N. Aronszajn. Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3):337–404, 1950.

S.-K. Au and J. L. Beck. Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic engineering mechanics*, 16(4):263–277, 2001.

S. Ba and V. R. Joseph. Composite Gaussian process models for emulating expensive functions. *The Annals of Applied Statistics*, 6(4):1838 – 1860, 2012.

F. Bachoc. *Estimation paramétrique de la fonction de covariance dans le modèle de krigeage par processus gaussiens: application à la quantification des incertitudes en simulation numérique*. PhD thesis, Paris 7, 2013a.

F. Bachoc. Cross validation and maximum likelihood estimation of hyper-parameters of Gaussian processes with model misspecification. *Computational Statistics and Data Analysis*, 2013b.

F. Bachoc. Asymptotic analysis of covariance parameter estimation for Gaussian processes in the misspecified case. *Bernoulli*, 24(2):1531–1575, 2018.

F. Bachoc and A. Lagnoux. Posterior contraction rates for constrained deep Gaussian processes in density estimation and classication. *arXiv preprint arXiv:2112.07280*, 2021.

F. Bachoc, A. Lagnoux, and T. Nguyen. Cross-validation estimation of covariance parameters under fixed-domain asymptotics. *Journal of Multivariate Analysis*, 160, 2016.

J. Y. Bao, F. Ye, and Y. Yang. Screening effect in isotropic Gaussian processes. *Acta Mathematica Sinica*, 36(5), 2020.

S. Barthelmé, P.-O. Amblard, N. Tremblay, and K. Usevich. Gaussian process regression in the flat limit. *arXiv preprint arXiv:2201.01074*, 2022.

S. Basak, S. J. Petit, J. Bect, and E. Vazquez. Numerical issues in maximum likelihood parameter estimation for Gaussian process regression. In *7th International Conference on machine Learning, Optimization and Data science*, 2021.

M. Baudin, A. Dutfoy, B. Iooss, and A. L. Popelin. OpenTURNS: an industrial software for uncertainty quantification in simulation. In R. Ghanem, D. Higdon, and H. Owhadi, editors, *Handbook of Uncertainty Quantification*, pages 2001–2038. Springer, Switzerland, 2017.

M. J. Bayarri, J. O. Berger, R. Paulo, J. Sacks, J. A. Cafeo, J. Cavendish, C. H. Lin, and J. Tu. A framework for validation of computer models. *Technometrics*, 49(2):138–154, 2007.

J. Bect, E. Vazquez, et al. STK: a Small (Matlab/Octave) Toolbox for Kriging. Release 2.6. http://kriging.sourceforge.net, 2011–2021.

J. Bect, D. Ginsbourger, L. Li, V. Picheny, and E. Vazquez. Sequential design of computer experiments for the estimation of a probability of failure. *Statistics and Computing*, 22(3):773–793, 2012.

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

R. Benassi, J. Bect, and E. Vazquez. Robust Gaussian process-based global optimization using a fully Bayesian expected improvement criterion. In Coello-Coello C. A., editor, *Learning and Intelligent Optimization, 5th International Conference, LION 5*, pages 176–190, Rome, Italy, January 17-21 2011. Springer.

R. Benassi, J. Bect, and E. Vazquez. Bayesian optimization using sequential Monte Carlo. In *Learning and Intelligent Optimization – 6th International Conference, LION 6*, pages 339–342, Paris, France, January 16-20 2012. Springer.

J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl. Algorithms for Hyper-Parameter Optimization. In *Advances in Neural Information Processing Systems 24 (NIPS 2011)*, volume 24. Curran Associates, Inc., 2011.

A. Berlinet and C. Thomas-Agnan. *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science & Business Media, 2004.

J. M. Bernardo. Expected information as expected utility. *The Annals of Statistics*, 7(3):686–690, 1979.

M. C. Bernardo, R. Buck, L. Liu, W. A. Nazaret, J. Sacks, and W. J. Welch. Integrated circuit design optimization using a sequential strategy. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 11(3): 361–372, 1992.

E. Bodin, M. Kaiser, I. Kazlauskaite, Z. Dai, N. Campbell, and C. H. Ek. Modulating surrogates for Bayesian optimization. In *International Conference on Machine Learning*, pages 970–979. PMLR, 2020.

H. Brezis. *Functional analysis, Sobolev spaces and partial differential equations*, volume 2. Springer, 2011.

A. D. Bull. Convergence rates of efficient global optimization algorithms. *Journal of Machine Learning Research*, 12(10), 2011.

R. Byrd, P. Lu, J. Nocedal, and C. Zhu. A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Comput.*, 16(5):1190–1208, 1995.

B. Calderhead, M. Girolami, and N. D. Lawrence. Accelerating Bayesian inference over nonlinear differential equations with Gaussian processes. In *Advances in Neural Information Processing Systems 21 (NIPS 2008)*, pages 217–224, 2009.

D. Chafekar, J. Xuan, and K. Rasheed. Constrained multi-objective optimization using steady state genetic algorithms. In *Genetic and Evolutionary Computation — GECCO 2003*, pages 813–824, Berlin, Heidelberg, 2003. Springer.

C. Chevalier and D. Ginsbourger. Fast computation of the multi-points expected improvement with applications in batch selection. In *International Conference on Learning and Intelligent Optimization*, pages 59–69. Springer, 2013.

C. Chevalier, J. Bect, D. Ginsbourger, E. Vazquez, V. Picheny, and Y. Richet. Fast parallel kriging-based stepwise uncertainty reduction with application to the identification of an excursion set. *Technometrics*, 56(4):455–465, 2014.

W. Chu and Z. Ghahramani. Gaussian processes for ordinal regression. *Journal of machine learning research*, 6(7), 2005.

A. R. Conn, N. I. M. Gould, and P. L. Toint. *Trust region methods*. SIAM, 2000.

P. Craven and G. Wahba. Smoothing noisy data with spline functions. Estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik*, 1979.

N. Cressie. *Statistics for Spatial Data*. Wiley, New York, 1993.

N. Cristianini, J. Shawe-Taylor, A. Elisseeff, and J. Kandola. On kernel-target alignment. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, NIPS'01, Cambridge, MA, USA, 2001. MIT Press.

C. Currin, T. Mitchell, M. Morris, and D. Ylvisaker. A Bayesian approach to the design and analysis of computer experiments. Technical report, Oak Ridge National Lab., TN (USA), 1988.

C. Currin, T. Mitchell, M. Morris, and D. Ylvisaker. Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments. *Journal of the American Statistical Association*, 86(416):953–963, 1991.

F. E. Curtis and X. Que. A quasi-Newton algorithm for nonconvex, nonsmooth optimization with global convergence guarantees. *Math. Program. Comput.*, 7 (4):399–428, 2015.

S. Da Veiga and A. Marrel. Gaussian process modeling with inequality constraints. In *Annales de la Faculté des sciences de Toulouse: Mathématiques*, volume 21, pages 529–555, 2012.

A. Damianou and N. D. Lawrence. Deep Gaussian processes. In *Artificial intelligence and statistics*, pages 207–215. PMLR, 2013.

C. Demay, B. Iooss, L. Le Gratiet, and A. Marrel. Model selection for Gaussian process regression: an application with highlights on the model variance validation. 2021. URL https://hal.archives-ouvertes.fr/hal-03207216.

J. L. Deutsch and C. V. Deutsch. Latin hypercube sampling with multidimensional uniformity. *Journal of Statistical Planning and Inference*, 142(3):763–772, 2012.

E. Di Nezza, G. Palatucci, and E. Valdinoci. Hitchhiker's guide to the fractional Sobolev spaces. *Bulletin des sciences mathématiques*, 136(5):521–573, 2012.

L. C. W. Dixon and G. P. Szegö. The global optimisation problem: An introduction. In L. C. W. Dixon and G. P. Szegö, editors, *Towards Global Optimisation 2*, pages 1–15. North Holland, Amsterdam, 1978.

O. Dubrule. Cross validation of kriging in a unique neighborhood. *Journal of the International Association for Mathematical Geology*, 15:687–699, 1983.

M. M. Dunlop, M. A. Girolami, A. M. Stuart, and A. L. Teckentrup. How deep are deep Gaussian processes? *Journal of Machine Learning Research*, 19(54): 1–46, 2018.

M. T. M. Emmerich, K. C. Giannakoglou, and B. Naujoks. Single- and multiobjective evolutionary optimization assisted by Gaussian random field metamodels. *IEEE Trans. Evol. Comput.*, 10(4):421–439, 2006.

C. B. Erickson, B. E. Ankenman, and S. M. Sanchez. Comparison of Gaussian process modeling software. *Eur. J. Oper. Res.*, 266(1):179–192, 2018.

D. Eriksson, M. Pearce, J. R. Gardner, R. C. Turner, and M. Poloczek. Scalable global optimization via local Bayesian optimization. In *NeurIPS*, 2019.

G. E. Fasshauer. Positive definite kernels: Past, present and future. *Dolomite Res. Notes Approx.*, 4, 2011.

G. E. Fasshauer, F. J. Hickernell, E. Lamie, P. Mokhasi, and J. Tate. "optimal" scaling and stable computation of meshfree kernel methods. Presented at the 3rd Workshop on High-Dimensional Approximation (HDA09), February 16–20, 2009, University of New South Wales, Sydney, Australia, 2009. URL http://math.iit.edu/~fass/FasshauerSydney.pdf.

P. Feliot. *Une approche Bayésienne pour l'optimisation multi-objectif sous contraintes*. PhD thesis, Université Paris-Saclay, 2017.

P. Feliot, J. Bect, and E. Vazquez. A Bayesian approach to constrained single-and multi-objective optimization. *Journal of Global Optimization*, 67(1-2):97–133, 2017.

A. Forrester, A. Sobester, and A. Keane. *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons, 2008.

P. I. Frazier, W. B. Powell, and S. Dayanik. A knowledge-gradient policy for sequential information collection. *SIAM Journal on Control and Optimization*, 47(5):2410–2439, 2008.

M. Fuhrländer and S. Schöps. A blackbox yield estimation workflow with Gaussian process regression applied to the design of electromagnetic devices. *Journal of Mathematics in Industry*, 10, 2020.

J. R. Gardner, G. Pleiss, D. Bindel, K. Q. Weinberger, and A. G. Wilson. GPyTorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration. In *Advances in Neur. Inform. Processing Systems*, volume 31. Curran Assoc., 2018.

M. Gaviano, D. E. Kvasov, D. Lera, and Y. D. Sergeyev. Algorithm 829: Software for generation of classes of test functions with known local and global minima for global optimization. *ACM Trans. Math. Softw.*, 29(4):469–480, 2003.

C. J. Geoga, O. Marin, M. Schanen, and M. L. Stein. Fitting Matérn smoothness parameters using automatic differentiation. *arXiv preprint arXiv:2201.00090*, 2022.

M. N. Gibbs. *Bayesian Gaussian processes for regression and classification*. PhD thesis, Citeseer, 1998.

D. Ginsbourger and R. Le Riche. Towards Gaussian process-based optimization with finite time horizon. In *mODa 9–Advances in Model-Oriented Design and Analysis*, pages 89–96. Springer, 2010.

D. Ginsbourger and C. Schärer. Fast calculation of Gaussian process multiple-fold cross-validation residuals and their covariances. *arXiv preprint arXiv:2101.03108*, 2021.

D. Ginsbourger, R. Le Riche, and L. Carraro. Kriging is well-suited to parallelize optimization. In *Computational intelligence in expensive optimization problems*, pages 131–162. Springer, 2010.

T. Gneiting and R. Ranjan. Comparing density forecasts using threshold-and quantile-weighted scoring rules. *Journal of Business & Economic Statistics*, 29(3):411–422, 2011.

T. G. Gneiting and A. E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102:359–378, 2007.

T. G. Gneiting, A. E. Raftery, A. H. Westveld III, and T. Goldman. Calibrated probabilistic forecasting using ensemble model output statistics and minimum crps estimation. *Monthly Weather Review*, 133(5):1098–1118, 2005.

G. Golub, M. Heath, and G. Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21:215–223, 1979.

R. B. Gramacy and H. K. H. Lee. Bayesian treed Gaussian process models with an application to computer modeling. *Journal of the American Statistical Association*, 103(483):1119–1130, 2008.

P. Grisvard. *Elliptic problems in nonsmooth domains*. Pitman, 1985.

M. S. Handcock and M. L. Stein. A Bayesian analysis of kriging. *Technometrics*, 35(4):403–410, 1993.

N. Hansen, S. Finck, R. Ros, and A. Auger. Real-Parameter Black-Box Optimization Benchmarking 2009: Noiseless Functions Definitions. Research Report RR-6829, INRIA, 2009.

N. Hansen, A. Auger, O. Mersmann, T. Tušar, and D. Brockhoff. Coco: A platform for comparing continuous optimizers in a black-box setting. *Optimization Methods and Software*, 36(1):114–144, 2021.

A. Hebbal, L. Brevault, and N. Melab. Bayesian optimization using deep Gaussian processes with applications to aerospace system design. *Optimization and Engineering*, 22(1):321–361, 2021.

A. R. Hedar and A. Ahmed. Studies on metaheuristics for continuous global optimization problems. 2004.

D. Higdon. A process-convolution approach to modelling temperatures in the north atlantic ocean. *Environmental and Ecological Statistics*, 5:173–190, 1998.

D. Higdon. Space and space-time modeling using process convolutions. *Quantitative Methods for Current Environmental Issues*, 04 2002.

K. Jakkala. Deep Gaussian processes: A survey. *arXiv preprint arXiv:2106.12135*, 2021.

D. Jones, M. Schonlau, and W. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13:455–492, 12 1998.

T. Karvonen. Asymptotic bounds for smoothness parameter estimates in Gaussian process regression. *arXiv preprint arXiv:2203.05400*, 2022.

T. Karvonen and C. J. Oates. Maximum likelihood estimation in Gaussian process regression is ill-posed. *arXiv preprint arXiv:2203.09179*, 2022.

T. Karvonen, G. Wynne, F. Tronarp, C. Oates, and S. Sarkka. Maximum likelihood estimation and uncertainty quantification for Gaussian process approximation of deterministic functions. *SIAM/ASA Journal on Uncertainty Quantification*, 8 (3):926–958, 2020.

M. C. Kennedy and A. O'Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3):425–464, 2001.

G. S. Kimeldorf and G. Wahba. A Correspondence Between Bayesian Estimation on Stochastic Processes and Smoothing by Splines. *The Annals of Mathematical Statistics*, 41(2):495 – 502, 1970.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In Y. Bengio and Y. LeCun, editors, *3rd Intern. Conf. on Learning Representations, ICLR 2015, San Diego, USA*, 2015.

P. K. Kitanidis. Statistical estimation of polynomial generalized covariance functions and hydrologic applications. *Water Resources Research*, 19(4):909–921, 1983.

M. Lázaro-Gredilla. Bayesian warped Gaussian processes. *Advances in Neural Information Processing Systems*, 25:1619–1627, 2012.

R. Le Riche and V. Picheny. Revisiting Bayesian Optimization in the light of the coco benchmark. *arXiv preprint arXiv:2103.16649*, 2021.

S. Lerch and T. L. Thorarinsdottir. Comparison of non-homogeneous regression models for probabilistic wind speed forecasting. *Tellus A: Dynamic Meteorology and Oceanography*, 65(1):21206, 2013.

J. R. Lewis, S. N. MacEachern, and Y. Lee. Bayesian restricted likelihood methods: Conditioning on insufficient statistics in Bayesian regression. *Bayesian Analysis*, 1(1), 2021.

S. Linnainmaa. The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors. Master's thesis, Univ. Helsinki, 1970.

D. Lizotte, R. Greiner, and D. Schuurmans. An experimental methodology for response surface optimization methods. *Journal of Global Optimization*, 53: 1–38, 08 2011.

A. F López-Lopera, F. Bachoc, N. Durrande, and O. Roustant. Finite-dimensional Gaussian approximation with linear inequality constraints. *SIAM/ASA Journal on Uncertainty Quantification*, 6(3):1224–1255, 2018.

H. Maatouk and X. Bay. Gaussian process emulators for computer experiments with inequality constraints. *Mathematical Geosciences*, 49(5):557–582, 2017.

S. Marmin, D. Ginsbourger, J. Baccou, and J. Liandrat. Warped Gaussian processes and derivative-based sequential designs for functions with heterogeneous variations. *SIAM/ASA Journal on Uncertainty Quantification*, 6(3):991–1018, 2018.

A. Marrel, B. Iooss, B. Laurent, and O. Roustant. Calculations of Sobol indices for the Gaussian process metamodel. *Reliability Engineering & System Safety*, 94(3):742–751, 2009.

J. D. Martin and T. W. Simpson. A study on the use of kriging models to approximate deterministic computer models. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages 567–576, 2003.

J. D. Martin and T. W. Simpson. Use of kriging models to approximate deterministic computer models. *AIAA Journal*, 2005.

B. Matérn. *Spatial Variation*. Springer-Verlag New York, 1986.

G. Matheron. The theory of regionalized variables and its applications. Technical Report Les cahiers du CMM de Fontainebleau, Fasc. 5, Ecole des Mines de Paris, 1971.

J. E. Matheson and R. L. Winkler. Scoring rules for continuous probability distributions. *Management science*, 22(10):1087–1096, 1976.

A. G. de G. Matthews, M. van der Wilk, T. Nickson, K. Fujii, A. Boukouvalas, P. León-Villagrá, Z. Ghahramani, and J. Hensman. GPflow: A Gaussian process library using TensorFlow. *J. Mach. Learn. Res.*, 18(40):1–6, 2017.

M. D. McKay, R. J. Beckman, and W. J. Conover. 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, 2000.

E. Meeds and S. Osindero. An alternative infinite mixture of Gaussian process experts. In *Advances in Neural Information Processing Systems 18 (NIPS 2005)*, pages 883–80. MIT Press, 2006.

E. Merrill, A. Fern, X. Z. Fern, and N. Dolatnia. An empirical study of Bayesian optimization: Acquisition versus partition. *The Journal of Machine Learning Research*, 22:4–1, 2021.

J. Mockus. On Bayesian methods for seeking the extremum. In G. I. Marchuk, editor, *Optimization Techniques IFIP Technical Conference Novosibirsk, July 1–7, 1974*, pages 400–404, Berlin, Heidelberg, 1975. Springer Berlin Heidelberg.

J. Mockus, V. Tiesis, and A. Zilinskas. *The application of Bayesian methods for seeking the extremum*, volume 2, pages 117–129. 1978.

S. Nadarajah and S. Kotz. Exact distribution of the max/min of two Gaussian random variables. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 16(2):210–212, 2008.

F. J. Narcowich, J. D. Ward, and H. Wendland. Sobolev error estimates and a bernstein inequality for scattered data interpolation via radial basis functions. *Constructive Approximation*, 24(2):175–186, 2006.

S. G. Nash. Newton-type minimization via the Lanczos method. *SIAM J. Numer. Anal.*, 21(4):770–788, 1984.

J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, USA, 2006a.

J. Nocedal and S. J. Wright. Quadratic programming. *Numerical optimization*, pages 448–492, 2006b.

D. J. Nott and W. T. M. Dunsmuir. Estimation of nonstationary spatial covariance structure. *Biometrika*, 89(4):819–829, 2002.

A. O'Hagan. Curve fitting and optimal design for prediction. *Journal of the royal statistical society series b-methodological*, 40:1–24, 1978.

C. Park and D. Apley. Patchwork kriging for large-scale Gaussian process regression. *The Journal of Machine Learning Research*, 19(1):269–311, 2018.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, 12:2825–2830, 2011.

C. Y. Peng and C. F. J. Wu. On the choice of nugget in kriging modeling for deterministic computer experiments. *Journal of Computational and Graphical Statistics*, 23(1):151–168, 2014.

S. J. Petit, J. Bect, S. Da Veiga, P. Feliot, and E. Vazquez. Presentation: optimisation bayésienne avec application à la conception d'un aubage fan. In *Séminaire EDF/CEA doctorants en statistiques et incertitudes*, 2019.

S. J. Petit, J. Bect, S. Da Veiga, P. Feliot, and E. Vazquez. Towards new cross-validation-based estimators for Gaussian process regression: efficient adjoint computation of gradients. In *52èmes Journées de Statistique de la SFdS (JdS 2020)*, 2020a.

S. J. Petit, J. Bect, P. Feliot, and E. Vazquez. Poster: Gaussian process model selection for computer experiments. In *Journées annuelles du GdR MASCOT NUM (MASCOT NUM 2020)*, 2020b.

S. J. Petit, J. Bect, P. Feliot, and E. Vazquez. Model parameters in Gaussian process interpolation: an empirical study of selection criteria. *Submitted to the SIAM/ASA Journal on Uncertainty Quantification*, 2021a. URL https://arxiv.org/abs/2107.06006.

S. J. Petit, J. Bect, P. Feliot, and E. Vazquez. Model parameters in Gaussian process interpolation: an empirical study of selection criteria: Supplementary Material. 2021b. URL https://hal-centralesupelec.archives-ouvertes.fr/hal-03285513v2/file/spetit-gpparam-suppmat.pdf.

S. J. Petit, J. Bect, and E. Vazquez. Presentation: Relaxed Gaussian process interpolation: a goal-oriented approach to Bayesian optimization. In *Journées annuelles du GdR MASCOT NUM (MASCOT NUM 2022)*, 2022a.

S. J. Petit, J. Bect, and E. Vazquez. Relaxed Gaussian process interpolation: a goal-oriented approach to Bayesian optimization. 2022b. URL https://arxiv.org/abs/2107.06006.

V. Picheny, D. Ginsbourger, O. Roustant, R. T. Haftka, and N. H. Kim. Adaptive designs of experiments for accurate approximation of a target region. *Journal of Mechanical Design*, 132(7), 2010.

V. Picheny, T. Wagner, and D. Ginsbourger. A benchmark of kriging-based infill criteria for noisy optimization. *Structural and multidisciplinary optimization*, 48 (3):607–626, 2013.

V. Picheny, S. Vakili, and A. Artemev. Ordinal Bayesian optimisation. *arXiv preprint arXiv:1912.02493*, 2019.

W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in C. The art of scientific computing*. Cambridge Univ. Press, 1992.

L. Pronzato and M. J. Rendas. Bayesian local kriging. *Technometrics*, 59(3): 293–304, 2017.

P. Ranjan, R. Haynes, and R. Karsten. A computationally stable approach to Gaussian process interpolation of deterministic computer simulation data. *Technometrics*, 53(4):366–378, 2011.

C. E. Rasmussen and Z. Ghahramani. Infinite mixtures of Gaussian process experts. *Advances in neural information processing systems*, 2:881–888, 2002.

C. E. Rasmussen and H. Nickisch. Gaussian processes for machine learning (GPML) toolbox. *J. Mach. Learn. Res.*, 11:3011–3015, 2010.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, USA, 2006.

R. G. Regis. Constrained optimization by radial basis function interpolation for high-dimensional expensive black-box problems with infeasible initial points. *Engineering Optimization*, 46(2):218–243, 2014.

J. Rivoirard and T. Romary. Continuity for kriging with moving neighborhood. *Mathematical Geosciences*, 43(4):469–481, 2011.

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

O. Roustant, D. Ginsbourger, and Y. Deville. Dicekriging, diceoptim: Two r packages for the analysis of computer experiments by kriging-based metamodeling and optimization. 2012b.

H. Rue and L. Held. *Gaussian Markov random fields: theory and applications*. Chapman and Hall/CRC, 2005.

I. Rychlik, P. Johannesson, and M. R. Leadbetter. Modelling and statistical analysis of ocean-wave data using transformed Gaussian processes. *Marine Structures*, 10(1):13–47, 1997.

J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn. Design and analysis of computer experiments. *Statistical science*, 4(4):409–423, 1989.

T. J. Santner, B. J. Williams, and W. I. Notz. *The Design and Analysis of Computer Experiments*. Springer series in statistics. Springer, 2003.

M. Scheurer, R. Schaback, and M. Schlather. Interpolation of spatial data – a stochastic or a deterministic problem? *European Journal of Applied Mathematics*, 24(4):601–629, 2013.

Sheffield machine learning group. GPy: A Gaussian process framework in Python, version 1.9.9. Available from http://github.com/SheffieldML/GPy, 2012–2020.

E. Snelson, C. E. Rasmussen, and Z. Ghahramani. Warped Gaussian processes. *Advances in neural information processing systems*, 16:337–344, 2004.

J. Snoek, H. Larochelle, and R. P. Adams. Practical Bayesian optimization of machine learning algorithms. In *25th Intern. Conf. on Neural Information Processing Systems. Volume 2*, pages 2951–2959. Curran Associates Inc., 2012.

N. Srinivas, A. Krause, S. M. Kakade, and M. Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *International Conference on Machine Learning*, pages 1015–1022, 2010.

M. L. Stein. Nonstationary spatial covariance functions. *Unpublished technical report*, 2005a.

M. L. Stein. Space-time covariance functions. *Journal of the American Statistical Association*, 100(469):310–321, 2005b.

M. L. Stein. 2010 rietz lecture: When does the screening effect hold? *The Annals of Statistics*, 39(6):2795–2819, 2011.

M.L. Stein. *Interpolation of Spatial Data: Some Theory for Kriging*. Springer Series in Statistics. Springer New York, 1999.

I. Steinwart, D. Hush, and C. Scovel. An explicit description of the reproducing kernel Hilbert spaces of Gaussian RBF kernels. *IEEE Transactions on Information Theory*, 52(10):4635–4643, 2006.

R. S. Strichartz. Multipliers on fractional Sobolev spaces. *Journal of Mathematics and Mechanics*, 16(9):1031–1060, 1967.

S. Sundararajan and S. S. Keerthi. Predictive approaches for choosing hyperparameters in Gaussian processes. *Neural Computation*, 13(5):1103–1118, 2001.

S. Surjanovic and D. Bingham. Virtual library of simulation experiments: Test functions and datasets. Retrieved October 13, 2020, from http://www.sfu.ca/~ssurjano/branin.html, 2013.

G. J. Székely and M. L. Rizzo. A new test for multivariate normality. *Journal of Multivariate Analysis*, 93(1):58–80, 2005.

L. N. Trefethen and D. Bau. *Numerical Linear Algebra*. SIAM, 1997.

V. Tresp. Mixtures of Gaussian processes. *Advances in neural information processing systems*, pages 654–660, 2001.

R. Tuo and W. Wang. Kriging prediction with isotropic matern correlations: Robustness and experimental designs. *Journal of Machine Learning Research*, 21 (187):1–38, 2020.

A. W. van Der Vaart and H. van Zanten. Information rates of nonparametric Gaussian process methods. *Journal of Machine Learning Research*, 12(6), 2011.

A. W. van der Vaart and J. H. van Zanten. Adaptive Bayesian estimation using a Gaussian random field with inverse gamma bandwidth. *The Annals of Statistics*, 37(5B):2655–2675, 2009.

J. Vanhatalo, J. Riihimäki, J. Hartikainen, P. Jylänki, V. Tolvanen, and A. Vehtari. Bayesian modeling with Gaussian processes using the MATLAB toolbox GPstuff (v3.3). *CoRR*, 2012.

E. Vazquez and J. Bect. Pointwise consistency of the kriging predictor with known mean and covariance functions. In *mODa 9–Advances in Model-Oriented Design and Analysis*, pages 221–228. Springer, 2010a.

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

E. Vazquez and J. Bect. A new integral loss function for Bayesian optimization. *arXiv preprint arXiv:1408.4622*, 2014.

J. Ver Hoef, N. Cressie, and R. Barry. Flexible spatial models for kriging and cokriging using moving averages and the fast fourier transform (fft). *Journal of Computational and Graphical Statistics - J COMPUT GRAPH STAT*, 13: 265–282, 2004.

J. Villemonteix, E. Vazquez, and E. Walter. An informational approach to the global optimization of expensive-to-evaluate functions. *Journal of Global Optimization*, 44(4):509–534, 2009.

P. Virtanen, R. Gommers, T. E. Oliphant, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

R. Von Mises. *Mathematical theory of probability and statistics*. Academic press, 1964.

G. Wahba. *Spline Models for Observational Data*. Society for Industrial and Applied Mathematics, Philadelphia, 1990.

G. Wahba and J. Wendelberger. Some new mathematical methods for variational objective analysis using splines and cross validation. *Monthly Weather Review*, 108:1122–1143, 1980.

H. Wendland. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in Computational Mathematics*, 4 (1):389–396, 1995.

H. Wendland. *Scattered data approximation*, volume 17. Cambridge university press, 2004.

R. Wilkinson. Accelerating ABC methods using Gaussian processes. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics (AISTATS 2014)*, volume 33 of *Proceedings of Machine Learning Research*, pages 1015–1023. PMLR, 2014.

B. A. Worley. Deterministic uncertainty analysis. Technical Report ORNL–6428, Oak Ridge National Laboratory, TN, USA, 1987.

G. Wynne, F.-X. Briol, and M. Girolami. Convergence guarantees for Gaussian process means with misspecified likelihoods and smoothness. *Journal of Machine Learning Research*, 22, 2021.

Y. Xiang, D. Y. Sun, W. Fan, and X. G. Gong. Generalized simulated annealing algorithm and its application to the Thomson model. *Physics Letters A*, 233(3): 216–220, 1997.

Y. Xiong, W. Chen, D. Apley, and X. Ding. A non-stationary covariance-based kriging method for metamodelling in engineering design. *International Journal for Numerical Methods in Engineering*, 71(6):733–756, 2007.

Y. Yang and J. Ma. An efficient em approach to parameter learning of the mixture of Gaussian processes. In *International Symposium on Neural Networks*, pages 165–174. Springer, 2011.

Z. Ying. Asymptotic properties of a maximum likelihood estimator with data from a Gaussian process. *J. Multivar. Anal.*, 36(2), 1991.

C. Yuan and C. Neubauer. Variational mixture of Gaussian process experts. In *Advances in Neural Information Processing Systems*, pages 1897–1904, 2009.

S. E. Yuksel, J. N. Wilson, and P. D. Gader. Twenty years of mixture of experts. *IEEE transactions on neural networks and learning systems*, 23(8):1177–1193, 2012.

H. Zhang. Inconsistent estimation and asymptotically equal interpolations in model-based geostatistics. *Journal of the American Statistical Association*, 99 (465):250–261, 2004.

H. Zhang and Y. Wang. Kriging and cross-validation for massive spatial data. *Environmetrics*, 21(3–4):290–304, 2010.