LEARNING GAUSSIAN PROCESSES WITH BAYESIAN POSTERIOR OPTIMIZATION

Luiz F. O. Chamon, Santiago Paternain, and Alejandro Ribeiro

Electrical and Systems Engineering, University of Pennsylvania
e-mail: {luizf,spater,aribeiro}@seas.upenn.edu

ABSTRACT

Gaussian processes (GPs) are often used as prior distributions in non-parametric Bayesian methods due to their numerical and analytical tractability. GP priors are specified by choosing a covariance function (along with its hyperparameters), a choice that is not only challenging in practice, but also has a profound impact on performance. This issue is typically overcome using hierarchical models, i.e., by learning a distribution over covariance functions/hyperparameters that defines a *mixture* of GPs. Yet, since choosing priors for hyperparameters can be challenging, maximum likelihood is often used instead to obtain point estimates. This approach, however, involves solving a non-convex optimization problem and is thus prone to overfitting. To address these issues, this work proposes a hybrid Bayesian-optimization solution in which the hyperparameters posterior distribution is obtained not using Bayes rule, but as the solution of a mathematical program. Explicitly, we obtain the hyperparameter distribution that minimizes a risk measure induced by the GP mixture. Previous knowledge, including properties such as sparsity and maximum entropy, is incorporated through (possibly non-convex) penalties instead of a prior. We prove that despite its infinite dimensionality and potential non-convexity, this problem can be solved exactly using duality and stochastic optimization.

1. INTRODUCTION

In practice, we are commonly faced with the challenge of extracting information from data and signals whose complexity is beyond that supported by existing parametric models. Simultaneously, it is often critical to determine the uncertainty associated with this information, especially when it is to be used as the basis for decision-making. Nonparametric Bayesian methods are particularly well-suited for these applications, since they provide distributions over function spaces conditioned on the observations. They can therefore be used to produce not only point estimates, such as those obtained from reproducing kernel Hilbert spaces (RKHSs) or splines models, but also uncertainty measures for those estimates. These techniques have been used to tackle problems from statistics, machine learning, control, and signal processing [1–5].

As in all of Bayesian inference, this distribution over functions—known as a *posterior*—is obtained by using Bayes' rule to combine a *likelihood*, arising from the measurement model, and a *prior*, for which Gaussian processes (GPs) have become a standard choice [2–4]. The success of GPs stems from their simplicity and flexibility. Indeed, though they are fully specified by the choice of a covariance function (along with its hyperparameters), there exists a large variety of admissible functions able to express smoothness, periodicity, and other structural properties. What is more, these functions can be combined to induce more complex properties on the solution [6]. Despite this flexibility, GPs remain numerically and analytically tractable, as their posterior can often be computed in closed-form [2].

The price for this flexibility is an additional burden to produce covariance functions appropriate for the task at hand. Indeed, this choice has a considerable effect on inference, making GP-based models susceptible to misspecification. In fact, even when observations abound, it can be hard for the evidence to overcome a misspecified GP prior in complex models [7–9]. This issue is aggravated by the limited prior knowledge available in many practical scenarios and by the difficulty in interpreting hyperparameters, especially when different covariance functions are combined.

In practice, the issue of selecting a covariance function is reduced to that of tuning a set of hyperparameters, either because the covariance structure is known or because one of the hyperparameters is used to select among families of covariance functions [2–4]. Two contrasting approaches are then used to learn these hyperparameters from data. The first obtains a point estimate by maximizing the likelihood of the observations with respect to the hyperparameters, a technique sometimes called type II maximum likelihood (ML) or ML-II. Though practical, the multimodal nature of the likelihood combined with the use of point estimates makes this approach prone to overfitting and unable to quantify the uncertainty associated with the hyperparameters [1, 2]. The second, more Bayesian approach is to place a prior on the hyperparameters and obtain a posterior distribution using traditional Monte Carlo Markov Chain (MCMC) methods. Though it addresses many of the issues from ML-II, the challenge of choosing priors for hyperparameters remains, either because their interpretation is not straightforward or because there is a lack of information as to what reasonable values are [6]. What is more, "noninformative" priors may have unexpected effects in hierarchical models, besides their numerical issues (e.g., improper posteriors) [2, 3, 10].

In this work, we propose a hybrid GP learning technique inspired by both Bayesian statistics and statistical optimization. As in Bayesian inference, we seek a distribution over the hyperparameters values instead of point estimates, reducing the risk of overfitting and allowing uncertainty to be quantified. Inspired by statistical learning, however, this hyperparameter distribution is obtaining not using Bayes' rule, but by minimizing a risk measure over the data. Hence, the posterior is obtained as the solution of an optimization problem without ever explicitly specifying a prior. Prior knowledge is incorporated using penalties terms that allow complex properties such as sparsity to be imposed. Though infinite dimensional and possibly non-convex, this optimization problem can be solved exactly using duality and stochastic optimization.

2. MODEL LEARNING FOR GPs

Let $(\boldsymbol{x}_i, y_i)$, $i = 1, \ldots, n$, be a set of independent observations where $\boldsymbol{x}_i \in \mathbb{R}^p$ and y_i is normally distributed according to

This work is supported by the ARL DCIST CRA W911NF-17-2-0181.

$$y_i \sim \mathcal{N}(f^o(\boldsymbol{x}_i), \sigma_\epsilon^2), \tag{1}$$

for unknown function f^o and variance σ_ϵ^2. The goal of GP estimation is to obtain a distribution over functions f conditioned on these observations. To do so, its leverages Bayes' rule to write

$$\mathbb{P}\left[f \mid \{\boldsymbol{x}_i, y_i\}_{i=1,\dots,n}\right] \propto \prod_{i=1}^{n} \mathbb{P}\left[y_i \mid \boldsymbol{x}_i, f\right] \mathbb{P}\left[f \mid \boldsymbol{x}_i\right], \tag{2}$$

where the likelihood $\mathbb{P}\left[y_i \mid \boldsymbol{x}_i, f\right] = \mathcal{N}(y_i \mid f(\boldsymbol{x}_i), \sigma_\epsilon^2)$ is obtained from (1) and the prior $\mathbb{P}\left[f \mid \boldsymbol{x}_i\right]$ is a GP. For convenience, $\boldsymbol{x}$ can be thought of as a feature vector or a system input, y as a label or a measurement, and f as a classifier or estimator.

A GP is a stochastic process whose finite dimensional marginals are multivariate Gaussians. Formally, let $m : \mathbb{R}^p \to \mathbb{R}$ be a *mean function* and $k : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$ positive-definite be a *covariance function*. Then, $\mathbb{GP}(m, k)$ is a distribution over functions g such that $\left[g(\boldsymbol{x}_1) \ \cdots \ g(\boldsymbol{x}_n)\right]^T \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{K})$ for all $n \in \mathbb{N}$ with $\boldsymbol{m} = \left[m(\boldsymbol{x}_1) \ \cdots \ m(\boldsymbol{x}_n)\right]^T$ and

$$\boldsymbol{K} = \begin{bmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & \cdots & k(\boldsymbol{x}_1, \boldsymbol{x}_n) \\ \vdots & \ddots & \vdots \\ k(\boldsymbol{x}_n, \boldsymbol{x}_1) & \cdots & k(\boldsymbol{x}_n, \boldsymbol{x}_n) \end{bmatrix}. \tag{3}$$

As is usual, we assume from now on that $m \equiv 0$ and simply write $\mathbb{GP}(k)$ [2]. The covariance function k often depends on hyperparameters that determine its properties, in which case we make the dependence explicit by writing $k_{\boldsymbol{\theta}}$, where $\boldsymbol{\theta} \in \mathcal{T}$ is a vector that collects the hyperparameters and $\mathcal{T} \subset \mathbb{R}^q$ is a compact set of *admissible* values. For instance, the commonly used squared exponential or radial basis function is given by

$$k_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 \exp\left[-\kappa \frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2}\right] + \sigma_\epsilon^2 \delta_{\boldsymbol{x}, \boldsymbol{x}'}, \tag{4}$$

where δ denotes the Kronecker delta. Here, $\boldsymbol{\theta} = \left[\sigma^2, \kappa, \sigma_\epsilon^2\right]^T$ determines the output scale, the length-scale, and the noise level.

The tractability of GPs comes from the fact that for measurements as in (1) and $(f \mid \{\boldsymbol{x}_i\}) \sim \mathbb{GP}(k)$, the posterior in (2) has a closed-form expression. Explicitly, for any point $\bar{\boldsymbol{x}} \in \mathbb{R}^n$,

$$(f(\bar{\boldsymbol{x}}) \mid \{\boldsymbol{x}_i, y_i\}) \sim \mathcal{N}\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right), \tag{5}$$

with $\boldsymbol{\mu} = \bar{\boldsymbol{k}}^T \boldsymbol{K}^{-1} \boldsymbol{y}$ and $\boldsymbol{\Sigma} = k(\bar{\boldsymbol{x}}, \bar{\boldsymbol{x}}) - \bar{\boldsymbol{k}}^T \boldsymbol{K}^{-1} \bar{\boldsymbol{k}}$, where $\boldsymbol{y}$ is a vector collecting the y_i from (1), $\bar{\boldsymbol{k}} = \left[k(\bar{\boldsymbol{x}}, \boldsymbol{x}_1) \ \cdots \ k(\bar{\boldsymbol{x}}, \boldsymbol{x}_n)\right]^T$, and $\boldsymbol{K}$ is as in (3). Notice that the Gram matrix $\boldsymbol{K}$ is fixed for a given covariance function k. As a result, its factorization can be computed *a priori*, reducing the cost of evaluating (5) to $\mathcal{O}(n^2)$ [2, 4].

It is clear from (5) that the inference result heavily depends on the choice of covariance function (and its hyperparameters). To sidestep the challenges involved in hand-picking these settings, hierarchical models are often used to include k, or more precisely its hyperparameters $\boldsymbol{\theta}$, in the inference procedure [2, 3, 10]. On the lower level, the distribution in (2) is conditioned on the hyperparameters to get

$$\mathbb{P}\left[f \mid \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\theta}\right] \propto \mathbb{P}\left[\boldsymbol{y} \mid \boldsymbol{X}, f, \boldsymbol{\theta}\right] \mathbb{P}\left[f \mid \boldsymbol{X}, \boldsymbol{\theta}\right]. \tag{6a}$$

The posterior is then evaluated by marginalizing over $\boldsymbol{\theta}$ as in

$$\mathbb{P}\left[f \mid \boldsymbol{X}, \boldsymbol{y}\right] = \int_{\mathcal{T}} \mathbb{P}\left[f \mid \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\theta}\right] \mathbb{P}\left[\boldsymbol{\theta} \mid \boldsymbol{X}, \boldsymbol{y}\right] d\boldsymbol{\theta}. \tag{6b}$$

Then, another level of inference is used to compute the hyperparameters posterior $\mathbb{P}\left[\boldsymbol{\theta} \mid \boldsymbol{X}, \boldsymbol{y}\right]$ required to evaluate (6b). Explicitly,

$$\mathbb{P}\left[\boldsymbol{\theta} \mid \boldsymbol{X}, \boldsymbol{y}\right] \propto \mathbb{P}\left[\boldsymbol{y} \mid \boldsymbol{X}, \boldsymbol{\theta}\right] \mathbb{P}\left[\boldsymbol{\theta} \mid \boldsymbol{X}\right] \tag{7a}$$

$$\begin{aligned} \mathbb{P}\left[\boldsymbol{y} \mid \boldsymbol{X}, \boldsymbol{\theta}\right] &= \int_{\mathbb{R}^p} \mathbb{P}\left[\boldsymbol{y} \mid \boldsymbol{X}, f(\bar{\boldsymbol{x}}), \boldsymbol{\theta}\right] \mathbb{P}\left[f(\bar{\boldsymbol{x}}) \mid \boldsymbol{X}, \boldsymbol{\theta}\right] d\bar{\boldsymbol{x}} \\ &= \frac{1}{\sqrt{(2\pi)^n \det(\boldsymbol{K}_{\boldsymbol{\theta}})}} \exp\left(-\frac{\boldsymbol{y}^T \boldsymbol{K}_{\boldsymbol{\theta}}^{-1} \boldsymbol{y}}{2}\right) \end{aligned} \tag{7b}$$

Notice that (6)–(7) enables learning the hyperparameters from the observations by leveraging the *hyper-prior* $\mathbb{P}\left[\boldsymbol{\theta} \mid \boldsymbol{X}\right]$. In other words, the hierarchical model has transferred the issue of choosing a prior from functions to hyperparameters. Still, we face similar interpretation and indeterminacy challenges as we did when selecting the covariance function of GPs [1, 2]. What is more, the use of noninformative priors can lead to numerical issues in hierarchical models [3] and given the underlying GP-based inference, we may again be confronted with misspecification issues [7–9]. Obtaining a point estimate for $\boldsymbol{\theta}$ to plug into (6a) by maximizing the likelihood in (7b) is also not without issues, since the objective is often multimodal (nonconvex), making the result prone to overfitting [1, 2].

In the sequel, we address these issues by taking advantage of the Bayesian hierarchical model (6)–(7) without explicitly specifying the hyperparameters prior in (7a). Inspired by statistical learning, we directly obtain the posterior $\mathbb{P}\left[\boldsymbol{\theta} \mid \boldsymbol{X}, \boldsymbol{y}\right]$, not by using Bayes' rule, but by minimizing an empirical risk over the distribution (6b). Desired properties and prior knowledge are encoded in penalties included in the in the objective. We then proceed to show that, despite its infinite dimensionality and possibly non-convex nature, the resulting optimization problem can be solved exactly under mild conditions.

3. BAYESIAN POSTERIOR OPTIMIZATION

We develop our approach by first explicitly formulating GP model learning as a statistical learning problem. Recall that in the latter, seek the function $\phi : \mathbb{R}^d \to \mathbb{R}$ in a function space $\mathcal{F}$ that minimizes the expected value of a risk measure $\ell : \mathbb{R}^2 \to \mathbb{R}_+$ over an unknown joint probability distribution $\mathcal{D}$ over data pairs $(\boldsymbol{x}, y)$. Explicitly,

$$\phi^\star \in \underset{\phi \in \mathcal{F}}{\operatorname{argmin}} \ \mathbb{E}_{(\boldsymbol{x}, y) \sim \mathcal{D}}\left[\ell(\phi(\boldsymbol{x}), y)\right] + R(\phi), \tag{PI}$$

where R is a penalty function used to describe prior knowledge or impose structure on $\phi^\star$ [11]. In contrast to (PI), model learning for GPs seeks not a function, but a distribution over functions (namely, a GP). Specifically, the function space $\mathcal{F}$ in (PI) is replaced by the space of GPs $\mathcal{GP}$ as in

$$\Gamma^\star \in \underset{\Gamma \in \mathcal{GP}}{\operatorname{argmin}} \ \mathbb{E}_{(\boldsymbol{x}, y) \sim \mathcal{D},\, f \sim \Gamma}\left[\ell(f(\boldsymbol{x}), y)\right] + R(\Gamma). \tag{PII}$$

Note that (PII) minimizes the expected value of ℓ not only over the unknown data distribution $\mathcal{D}$, but also over the GP Γ.

To solve (PII), we must overcome three challenges of statistical, representational, and algorithmic natures. The first (statistical) is that we do not know $\mathcal{D}$ to evaluate the objective of (PII). This is a classical problem in statistical learning that is overcome using data. To be sure, we can use realizations $(\boldsymbol{x}_i, y_i) \sim \mathcal{D}$ to approximate the expectation by an empirical average as in

$$\hat{\Gamma}^\star \in \underset{\Gamma \in \mathcal{GP}}{\operatorname{argmin}} \ \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}_{f \sim \Gamma}\left[\ell(f(\boldsymbol{x}_i), y_i)\right] + R(\Gamma). \tag{P$\hat{\text{I}}$I}$$

Statistical learning theory is often concerned with the conditions under which $\hat{\Gamma}^\star$ is close to $\Gamma^\star$ [11]. The second challenge (representational) comes from the difficulty in optimizing over $\mathcal{GP}$. Indeed, the space of GPs is isomorphic to the space of positive-definite functions for which it is hard to obtain practical representations. Though approximations based on spectral representations have been proposed, they can be hard to optimize over due to the resulting non-convex mathematical programs [12, 13]. We overcome this issue by leveraging the marginal representation in (6b) to write

$$p^\star \in \underset{p \in \mathcal{P}}{\operatorname{argmin}} \ \frac{1}{n} \sum_{i=1}^{n} \int_{\mathcal{T}} \mathbb{E}_{f \sim \mathbb{GP}(k_{\boldsymbol{\theta}})} \left[\ell(f(\boldsymbol{x}_i), y_i) \right] p(\boldsymbol{\theta}) d\boldsymbol{\theta} + R(p), \tag{PIII}$$

where $\mathcal{P}$ is the space of probability densities. Note that the remaining expected value in the objective of (PIII) is a Gaussian integral completely defined by $k_{\boldsymbol{\theta}}$. It can therefore be computed efficiently using Gauss-Hermite quadrature or in the case of quadratic losses, even be obtained in closed-form. The GP is then obtained by marginalizing over $\boldsymbol{\theta}$ as in

$$\hat{\Gamma}_p^\star = \int_{\mathcal{T}} \mathbb{P}\left(f \mid \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\theta}\right) p^\star(\boldsymbol{\theta}) d\boldsymbol{\theta}. \tag{8}$$

Observe from (8) that $p^\star$ takes the place of the hyperparameter posterior $\mathbb{P}\left(\boldsymbol{\theta} \mid \boldsymbol{X}, \boldsymbol{y}\right)$ from (6b). Since its optimization variable is in fact a posterior distribution, we call (PIII) a *Bayesian posterior optimization problem*. Note, however, that no prior distribution exists in the context of (PIII): the posterior is obtained directly from data by solving an empirical risk minimization problem. That is not to say that prior knowledge cannot be incorporated through the penalty R. Indeed, this regularization can be used to promote structural properties of the posterior as well incorporate *a priori* information on the value of the hyperparameters. Examples involving entropy, sparsity, and moments are presented in Section 5.

The integral in (8) also lends itself to a parametric, non-Bayesian interpretation. Let $\mathfrak{D} = \{P(\cdot, \boldsymbol{\theta}) \in \mathcal{P}(\mathcal{F}) \mid \boldsymbol{\theta} \in \mathcal{T}\}$ be a dictionary that contains a continuum of probability densities P over the space of functions $\mathcal{F}$. We can then approximate Bayesian inference over $\mathcal{F}$ by the optimal coding problem of finding the convex combination of elements of $\mathfrak{D}$ that minimizes an empirical loss over the observations [14]. In a sense, $\mathfrak{D}$ is an infinite dimensional parametrization of (a subset of) the space of distributions over $\mathcal{F}$. In the case of (8), $\mathfrak{D}$ provides a parameterization of a subset of $\mathcal{GP}$ and (PIII) finds the GP in the span of $\mathfrak{D}$ that optimally fits the data $\{\boldsymbol{x}_i, y_i\}$.

Though readily useful, (PIII) is an infinite dimensional optimization problem that, depending on the choice of penalty R, could also be non-convex. This brings us to the final challenge (algorithmic) of obtaining a practical, efficient method to solve (PIII). In the next section, we show that despite its appearance of intractability, (PIII) can be solved exactly using duality under mild assumptions (most notably, still allowing non-convex penalties R).

4. SOLVING BAYESIAN POSTERIOR OPTIMIZATION PROBLEMS

To develop a simple and practical algorithm for solving (PIII), we leverage recent results from duality theory to write a convex optimization problem from which a solution of (PIII) can be recovered. This single variable mathematical program is then solved using the probabilistic bisection algorithm (PBA) [15, 16], though stochastic gradient ascent methods can also be used. Before proceeding with the derivations, we state the assumptions under which our algorithm is guaranteed to solve (PIII):

A.1 The penalty function R is a separable functional, i.e., the penalty is of the form

$$R = \int_{\mathcal{T}} h\left[p(\boldsymbol{\theta}), \boldsymbol{\theta}\right] d\boldsymbol{\theta} \tag{9}$$

A.2 The density $p^\star$ and the functions h and ℓ are non-atomic.

A.3 The risk ℓ in (PIII) and the function h in (9) are normal integrands and the objective of (PIII) is strongly convex.

Algorithm 1 Bayesian posterior optimization

```
For an upper ($\bar{\xi}$) and lower ($\underline{\xi}$) bound on $\xi^\star$, a numerical integration method $I$, and $s$ such that $\mathbb{P}\left[\operatorname{sign}(I(p) - 1) = \operatorname{sign}(\int p(\boldsymbol{\theta}) d\boldsymbol{\theta} - 1)\right] > s$, initialize $r_0(\xi) = (\bar{\xi} - \underline{\xi})^{-1}$ for $\xi \in [\underline{\xi}, \bar{\xi}]$ and zero otherwise
for $t = 0, \dots, T-1$
    $\xi_t \leftarrow$ median of $r_t$
    if $I(p_{\xi_t}) > 1$ then
        $r_{t+1}(\xi) = \begin{cases} 2(1-s) r_t(\xi), & \xi < \xi_t \\ 2s r_t(\xi), & \xi \geq \xi_t \end{cases}$
    else
        $r_{t+1}(\xi) = \begin{cases} 2s r_t(\xi), & \xi < \xi_t \\ 2(1-s) r_t(\xi), & \xi \geq \xi_t \end{cases}$
    end if
end
$p^\star(\boldsymbol{\theta}) = p_{\xi^\star}(\boldsymbol{\theta})$ for $\xi^\star = \operatorname{argmax}_{\xi} r_T(\xi)$
```

Assumptions 1 and 2 are used to prove that (PIII) can be solved using duality even if the penalty R is non-convex. Assumption 3 allows us to overcome the challenge of infinite dimensionality by solving optimizing p individually for each $\boldsymbol{\theta}$. Though the strong convexity assumption is not necessary, it simplifies the derivations as the solution of (PIII) then becomes unique [17].

First, notice that, under Assumptions 1–3, (PIII) is equivalent to

$$\begin{aligned} p^\star = \underset{p \in L_1^+}{\operatorname{argmin}} \quad & \frac{1}{n} \sum_{i=1}^{n} \int_{\mathcal{T}} \mathbb{E}_{f \sim \mathbb{GP}(k_{\boldsymbol{\theta}})} \left[\ell(f(\boldsymbol{x}_i), y_i) \right] p(\boldsymbol{\theta}) d\boldsymbol{\theta} \\ & + \sum_{r=1}^{C} \lambda_r \int_{\mathcal{T}} h_r \left[p(\boldsymbol{\theta}), \boldsymbol{\theta}\right] d\boldsymbol{\theta} \\ \text{subject to} \quad & \int_{\mathcal{T}} p(\boldsymbol{\theta}) d\boldsymbol{\theta} = 1, \end{aligned} \tag{PIV}$$

where the optimization is now performed over L_1^+, the space of integrable, non-negative valued functions, and we allow C different penalties weighted by $\lambda_r > 0$ for $r = 1, \dots, C$. Proceed by defining the Lagrangian associated with (PIV) as

$$\begin{aligned} \mathcal{L}(p, \xi) = & \frac{1}{n} \sum_{i=1}^{n} \int_{\mathcal{T}} \mathbb{E}_{f \sim \mathbb{GP}(k_{\boldsymbol{\theta}})} \left[\ell(f(\boldsymbol{x}_i), y_i) \right] p(\boldsymbol{\theta}) d\boldsymbol{\theta} \\ & + \sum_{r=1}^{C} \lambda_r \int_{\mathcal{T}} h_r \left[p(\boldsymbol{\theta}), \boldsymbol{\theta}\right] d\boldsymbol{\theta} + \xi \left[\int_{\mathcal{T}} p(\boldsymbol{\theta}) d\boldsymbol{\theta} - 1 \right], \end{aligned} \tag{10}$$

its dual function as $d(\xi) = \min_{p \in L_1^+} \mathcal{L}(p, \xi)$, and its dual problem as

$$\underset{\xi \in \mathbb{R}}{\operatorname{maximize}} \ d(\xi). \tag{DIV}$$

This dual problem is attractive for two reasons. First, it is a convex optimization problem. This is in fact true of all dual problems since the dual function is defined as a minimum of affine functions and must therefore be concave [17]. In practice, this implies that if we can evaluate $d(\xi)$, we can solve (DIV). The second reason is laid out in the following theorem that shows that the solutions of (PIV) and (DIV) are intrinsically connected:

Theorem 1. *Let $\xi^\star$ be any solution of* (DIV). *Then, under Assumptions 2 and 3, it holds that*

$$p^\star = \underset{p \in L_1^+}{\operatorname{argmin}} \ \mathcal{L}(p, \xi^\star) \tag{11}$$

As opposed to the first point, this result is not trivial. It stems from the fact that $\mathcal{L}$ is strongly convex (Assumption 3) and that (PIV) is a particular case of a sparse functional program (SFP) [14]. Though non-convex, SFPs have been shown to have no duality gap, thus allowing them to be solved exactly using duality [14, Thm. 1]. Due to space constraints, we defer the formal proof of this result to the extended version of this work.

All that is left now is to compute the minimizer in (11). If this minimization is tractable, then the objective of (DIV) and consequently $\xi^\star$ can be computed efficiently. As a result, Theorem 1 implies that (PIII) itself can be solved efficiently. To solve the optimization problem (11), note that the Lagrangian (10) can be written as

$$\mathcal{L}(p, \xi) = \int_{\mathcal{T}} \left[\left(\bar{\ell}(\boldsymbol{\theta}) + \xi\right) p(\boldsymbol{\theta}) + \sum_{r=1}^{C} \lambda_r h_r \left[p(\boldsymbol{\theta}), \boldsymbol{\theta}\right] \right] d\boldsymbol{\theta} - \xi \tag{12}$$

with

$$\bar{\ell}(\boldsymbol{\theta}) = \frac{1}{n} \sum_{i=1}^{n} \int_{\mathbb{R}} \ell(\hat{y}_i, y_i) \mathcal{N}\left(\hat{y}_i \mid \boldsymbol{\mu}_{\boldsymbol{\theta}}, \boldsymbol{\Sigma}_{\boldsymbol{\theta}}\right) d\hat{y}_i, \tag{13}$$

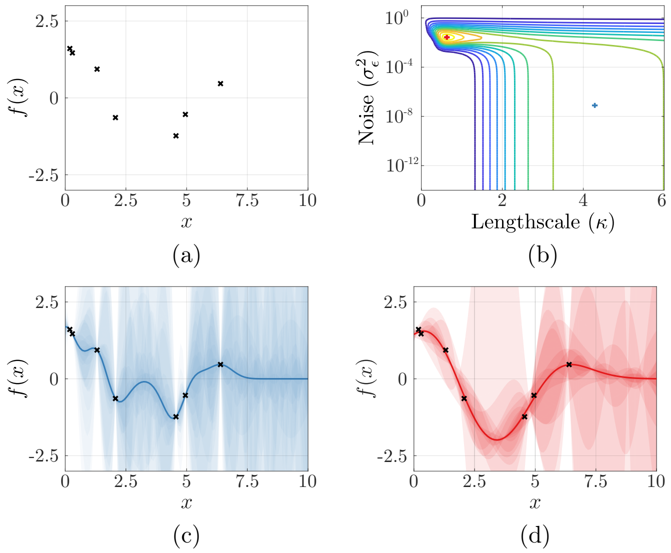
where we used (5) to replace the posterior $\mathbb{P}\left[f(\boldsymbol{x}_i) \mid \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\theta}\right]$ by a Gaussian distribution whose parameters are determined by the covariance function $k_{\boldsymbol{\theta}}$. Since ℓ and h_r are normal integrands and L_1^+ is a separable space, the minimum and the integral can exchanged [18, Thm. 3A]. Hence, we can solve the minimization individually for each $\boldsymbol{\theta}$ and obtain

$$p_\xi(\boldsymbol{\theta}) = \underset{p \geq 0}{\operatorname{argmin}} \left(\bar{\ell}(\boldsymbol{\theta}) + \xi\right) p + \sum_{r=1}^{C} \lambda_r h_r \left[p, \boldsymbol{\theta}\right]. \tag{14}$$

Even when the h_r are non-convex, this scalar problem often has a simple closed-form solution, as is the case for the negative entropy, sparsity, and first-order moment penalties using in Section 5. A step-by-step procedure to solve (PIII) is presented in Algorithm 1, where $\xi^\star$ is obtained using PBA [15, 16].

5. NUMERICAL EXAMPLE

To illustrate the use of Bayesian posterior optimization, we consider a scalar regression problem. We draw $n = 7$ points from a zero-mean GP with squared exponential covariance function as in (4) and hyperparameters $\sigma^2 = 1$, $\kappa = 1$, and $\sigma_\epsilon^2 = 0.1$ (Figure 1a). We assume that the output scale σ^2 is known and seek to learn the other two parameters, i.e., $\boldsymbol{\theta} = \left[\kappa,\ \sigma_\epsilon^2\right]^T$, over the ranges $\kappa \in [0, 6]$ and $\sigma_\epsilon^2 \in [10^{-14}, 1]$. Figure 1b shows the likelihood of the observations with respect to the remaining hyperparameters, i.e., $\mathbb{P}\left[\boldsymbol{y} \mid \boldsymbol{X}, \kappa, \sigma_\epsilon^2\right]$. Notice that it has two local maxima: one corresponds to a less smooth and less noisy solution (blue mark and Figure 1c) and the other corresponds to a smoother and more noisy explanation of the data (red mark and Figure 1d). The predictive curves are obtained using the learned GPs to fit 200 equally spaced points $\bar{x}$ in the range $[0, 10]$. Since these predictions are multivariate Gaussian distributions, we use transparency to plot each of the 200 principal axes of their 95% confidence ellipsoid. Notably, these observations do not support rejecting either of these hypotheses, illustrating why ML-II (or any other point estimate method) are susceptible to overfitting.



(a)

(b)

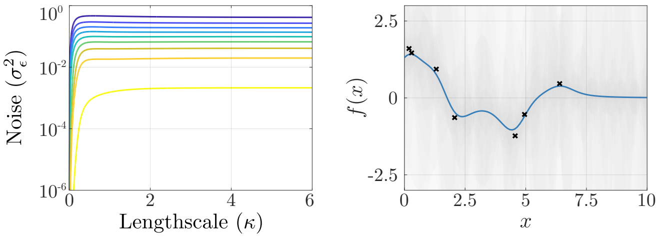
(c)

(d)

Fig. 1. Data description: (a) observations; (b) likelihood of observations with respect to the hyperparameters; (c) local maximum (non-smooth, low noise); (d) local maximum (smooth, high noise).

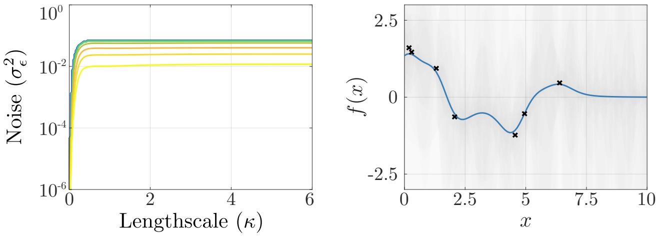
To proceed, we solve (PIII) for the classical squared loss $\ell(z, z') = (z - z')^2$ with a negative entropy penalty $h_1(z) = z \log(z)$ and $\lambda_1 = 1$. Notice from Figure 2a, that the resulting hyperparameter posterior suggests that, though the data do not support noise levels above 0.1, they cannot distinguish between a wide range of lengthscales κ. The result is that the mean of the learned GP displays an intermediate smoothness between Figures 1c and 1d, but its confidence intervals support different degrees of smoothness (Figure 2b). In order to prune low probability regions of the hyperparameter posterior, we may wish to trade-off entropy and sparsity by adding a penalty of the form $h_2(z) = \mathbb{I}(z \neq 0)$, where $\mathbb{I}(z \neq 0) = 1$ if $z \neq 0$ and zero otherwise. Figure 3 shows the result of doing so for $\lambda_2 = 2$. By enforcing sparsity on the posterior, it has now become clear that not only would GPs with high noise not fit the observations, but neither would GPs with very small lengthscales. Despite the non-convexity of h_2, Theorem 1 guarantees that Figure 3a is indeed $p^\star$. Finally, if prior knowledge about the problem suggests that the underlying function is smooth, we can encode this information in (PIII) by adding a penalty on the first-order moment of κ, i.e., taking $h_3\left[p(\boldsymbol{\theta}), \boldsymbol{\theta}\right] = \kappa p(\boldsymbol{\theta})$. In Figure 4, we display the hyperparameter posterior and predictive curve for $\lambda_3 = 1$. Though the resulting $p^\star$ concentrates around small lengthscales, there are still GPs with a wide range of noise levels that would fit the data.

Rather than explicitly inducing GPs with small lengthscales, we can replace the traditional squared loss by a cross-validated (CV) version using leave-one-out. Namely, for each i, we can compute the $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$ used to evaluate $\bar{\ell}$ in (13) without $(\boldsymbol{x}_i, y_i)$. The empirical loss then becomes the average *prediction* error of the GP on the dataset. Reverting to $\lambda_3 = 0$ and choosing $\lambda_1 = 5$ and $\lambda_2 = 3$ yields Figure 5. Now, only smoother GPs fit the observations and the posterior in Figure 5a displays an accumulation of mass around $\kappa = 0.3$ and $\sigma_\epsilon^2 = 10^{-3}$. Still, the noise level remains uncertain due to the small number of observations. Also note that the CV loss suggests the data could also be predicted by degenerate GPs with small lengthscales and noise, i.e., by effectively deterministic, constant functions.



(a) (b)

Fig. 2. Bayesian posterior optimization with ℓ_2 loss and negative entropy penalty: (a) $p^\star$ and (b) prediction



(a) (b)

Fig. 3. Bayesian posterior optimization with ℓ_2 loss and negative entropy and sparsity penalties: (a) $p^\star$ and (b) prediction

6. CONCLUSION

We addressed the issue of learning a GP prior for nonparametric Bayesian regression using a hybrid technique that, as in Bayesian inference, seeks a distribution over the hyperparameters values, but inspired by statistical learning, obtains this distribution not using Bayes' rule, but by minimizing a risk measure over the data. A hyperparameter posterior is then obtained as the solution of an optimization problem without ever explicitly specifying a prior. *A priori* information can be incorporated using (possibly non-convex) penalties terms such as entropy and sparsity. Though infinite dimensional and possibly non-convex, we show that this optimization problem can be solved exactly using duality and stochastic optimization. We believe this technique may be used beyond GPs to address the issue of learning priors and systematize Bayesian modeling procedures.

7. REFERENCES

[1] M.L. Stein, Ed., *Interpolation of Spatial Data: Some Theory for Kriging*, Springer, 1999.

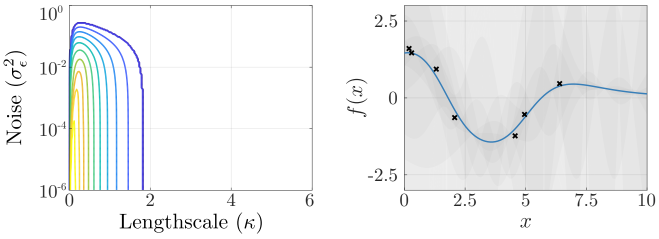
[2] C.E. Rasmussen and C.K.I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2005.

[3] A. Gelman, J.B. Carlin, H. S. Stern, D.B. Dunson, A. Vehtari, and D.B. Rubin, *Bayesian Data Analysis*, CRC Press, 2013.

[4] F. Pérez-Cruz, S. Van Vaerenbergh, J.J. Murillo-Fuentes, M. Lázaro-Gredilla, and I. Santamaría, "Gaussian processes for nonlinear signal processing: An overview of recent advances," *IEEE Signal Process. Mag.*, vol. 30[4], pp. 40–50, 2013.

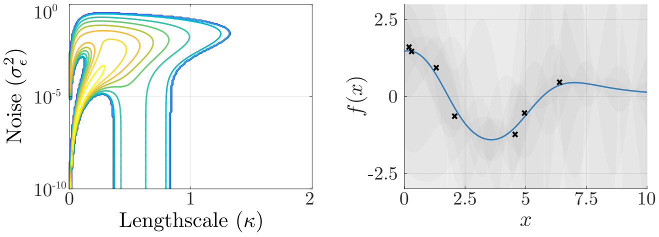
[5] M. Liu, G. Chowdhary, B. Castra da Silva, S. Liu, and J.P. How, "Gaussian processes for learning and control: A tutorial with examples," *IEEE Control Syst. Mag.*, vol. 38[5], pp. 53–86, 2018.

[6] D. Duvenaud, J. Lloyd, R. Grosse, J. Tenenbaum, and G. Zoubin, "Structure discovery in nonparametric regression through compositional kernel search," in *ICML*, 2013, pp. 1166–1174.



(a) (b)

Fig. 4. Bayesian posterior optimization with ℓ_2 loss and negative entropy, sparsity, and mean of κ penalties: (a) $p^\star$ and (b) prediction



(a) (b)

Fig. 5. Bayesian posterior optimization with CV ℓ_2 loss and negative entropy and sparsity penalties: (a) $p^\star$ and (b) prediction

[7] François Bachoc, "Cross validation and maximum likelihood estimations of hyper-parameters of Gaussian processes with model misspecification," *Computational Statist. & Data Anal.*, vol. 66, pp. 55–69, 2013.

[8] T. Beckers, J. Umlauft, and S. Hirche, "Mean square prediction error of misspecified Gaussian process models," in *Conf. on Decision and Control*, 2018, pp. 1162–1167.

[9] A. Zaytsev, E. Romanenkova, and D. Ermilov, "Interpolation error of Gaussian process regression for misspecified case," in *Workshop on Conformal and Probabilistic Prediction and Appl.*, 2018, pp. 83–95.

[10] C.E. Rasmussen and Z. Ghahramani, "Infinite mixtures of Gaussian process experts," in *NIPS*, 2001, pp. 881–888.

[11] V. Vapnik, *The nature of statistical learning theory*, Springer, 2013.

[12] A.G. Wilson and R.P. Adams, "Gaussian process kernels for pattern discovery and extrapolation," in *ICML*, 2013, pp. III–1067–III–1075.

[13] S. Remes, M. Heinonen, and S. Kaski, "Non-stationary spectral kernels," in *NIPS*, 2017, pp. 4642–4651.

[14] L.F.O. Chamon, Y.C. Eldar, and A. Ribeiro, "Functional nonlinear sparse models," *IEEE Trans. Signal Process. (submitted)*, 2018, `https://arxiv.org/abs/1811.00577`.

[15] M. Horstein, "Sequential transmission using noiseless feedback," *IEEE Trans. Inf. Theory*, vol. 9[3], pp. 136–143, 1963.

[16] P.I. Frazier, S.G. Henderson, and R. Waeber, "Probabilistic bisection converges almost as quickly as stochastic approximation," *Mathematics of Operations Research*, vol. 44[2], pp. 651–667, 2019.

[17] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge University Press, 2004.

[18] R. T. Rockafellar, *Integral functionals, normal integrands and measurable selections*, Springer, 1976.