Kernelized Normalizing Constant Estimation: Bridging Bayesian Quadrature and Bayesian Optimization

Xu Cai[1], **Jonathan Scarlett**[1,2]

[1] Department of Computer Science, National University of Singapore
[2] Department of Mathematics, Institute of Data Science, National University of Singapore
caix@u.nus.edu, scarlett@comp.nus.edu.sg

Abstract

In this paper, we study the problem of estimating the normalizing constant through queries to the black-box function f, which is the integration of the exponential function of f scaled by a problem parameter *lambda*. We assume f belongs to a *reproducing kernel Hilbert space* (RKHS), and show that to estimate the normalizing constant within a small relative error, the level of difficulty depends on the value of *lambda*: When *lambda* approaches zero, the problem is similar to *Bayesian quadrature* (BQ), while when *lambda* approaches infinity, the problem is similar to *Bayesian optimization* (BO). More generally, the problem varies between BQ and BO. We find that this pattern holds true even when the function evaluations are noisy, bringing new aspects to this topic. Our findings are supported by both algorithm-independent lower bounds and algorithmic upper bounds, as well as simulation studies conducted on a variety of benchmark functions.

1 Introduction

The problem of *normalizing constant (NC) estimation* (also known as *(log-)partition function estimation*) is of interest in a variety fields, such as Bayesian statistics (Chen and Shao 1997; Gelman and Meng 1998), machine learning (Desjardins, Bengio, and Courville 2011), statistical mechanics (Stoltz, Rousset et al. 2010), and other areas involving the distribution of an energy function. Given a distribution on a domain D with measure $d\mathbf{x}$, the normalizing constant is the integral $Z = \int_D e^{-f(\mathbf{x})} d\mathbf{x}$. In many classical works, f is assumed to be (strongly) convex, so that the distribution is (strongly) log-concave. Asymptotic/non-asymptotic performance bounds for the log-concave setting have been studied in detail; see for example (Ge, Lee, and Lu 2020) and the references therein.

Although the first-order (gradient) information of f is usually assumed to be available in the classical setting, it is also of interest to estimate the normalizing constant of non-convex black-box functions using only zeroth-order bandit feedback. Following the rich literature on Bayesian optimization (BO) and Bayesian quadrature (BQ), it is particularly natural to consider functions lying in a *reproducing kernel Hilbert space* (RKHS), for which Gaussian process (GP) techniques can be used to quantify uncertainty. The NC estimation problem brings new challenges compared to BO (which seeks to find $\arg\max_{\mathbf{x}\in D} f(\mathbf{x})$) and BQ (which seeks to approximate $\int_D f(\mathbf{x})d\mathbf{x}$), and in fact turns out to share features of both.

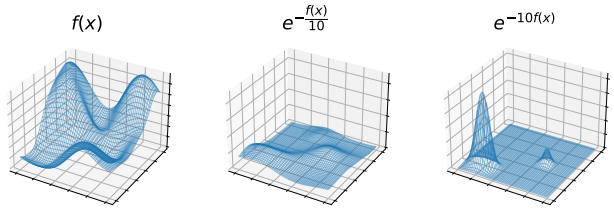


Figure 1: An illustration of estimating the normalizing constant under small (0.1) and large (10) λ.

In this paper, we study the asymptotic limits of NC estimation for functions in RKHS with bounded norm, adopting Bayesian numerical analysis techniques. As is commonly done, we consider an additional parameter λ in the estimation problem, i.e., we consider $Z = \int_D e^{-\lambda f(\mathbf{x})} d\mathbf{x}$. The interpretation of λ differs across different subjects; for instance, it can represent the reciprocal of the thermodynamic temperature of a system, or it can be used to "temper" or "amplify" terms in Bayesian statistics. Additionally, we consider the scenario where observations may be corrupted by additive Gaussian noise with variance σ^2, and λ and σ may vary with the time horizon T as $T \to \infty$.

For d-dimensional Matérn-ν RKHS functions, our findings reveal that, to estimate Z within a multiplicative factor of $1 \pm \epsilon$ with constant probability (e.g., 0.99), the level of difficulty generally exhibits the following behaviour (also partially listed in Table 1):

- When $\lambda \to 0$, the error bound of ϵ is similar to BQ. Our lower bound is similar to the BQ lower bounds stated in (Plaskota 1996; Cai, Lam, and Scarlett 2023), who showed that the order-optimal average *mean absolute error* of ϵ is $\Theta(T^{-\frac{\nu}{d}-1} + \sigma T^{-\frac{1}{2}})$.
- When $\lambda \to \infty$, the problem becomes more similar to BO, and the error bounds again reflect this. For instance, when $\lambda = \Theta(T)$ and $T \to \infty$, our noiseless $\Omega(T^{-\frac{\nu}{d}})$ lower

$\sigma = 0$ (noiseless)			$\sigma = \Theta(T^{-\frac{1}{4}})$	
$\lambda \to 0$	$\lambda \to \infty$		$\lambda \to 0$	$\lambda \to \infty$
$\Theta(\lambda T^{-\frac{\nu}{d}-1})$	$\Omega(\lambda T^{-\frac{\nu}{d}-1})$	$O(\lambda T^{-\frac{\nu}{d}-\frac{1}{2}})$	$\Theta(\lambda T^{-\frac{3}{4}})$ (needs $\nu \geq \frac{d}{2}$ when $\lambda \to \infty$)	

$\sigma = \Theta(T^{-\frac{1}{2}})$			$\sigma = \Theta(1)$ (constant)		
$\lambda \to 0$	$\lambda \to \infty$		$\lambda \to 0$	$\lambda \to \infty$	
$\Theta(\lambda T^{-1})$	$\Omega(\lambda T^{-1})$	$O^*\big(\lambda T^{-\frac{4\nu+d}{4\nu+2d}}\big)$	$\Theta(\lambda T^{-\frac{1}{2}})$	$\Omega(\lambda T^{-\frac{1}{2}})$	$O^*\big(\lambda T^{-\frac{\nu}{2\nu+d}}\big)$

Table 1: Selected results instantiated from Theorems 1, 2, 3 and 4 for NC. $O^*(\cdot)$ hides $\mathrm{poly}(\log T)$ terms. $\Theta(\cdot)$ means the lower and upper bounds match up to constant factors. For $\lambda \to \infty$, we treat $\lambda = \Theta(T^c)$, $c > 0$ and $T \to \infty$.

bound coincides with the noiseless BO lower bound stated in (Bull 2011), who showed that the average *simple regret* is $\Theta(T^{-\frac{\nu}{d}})$. For the noisy setting, the results and their tightness vary depending on the noise level. When $\sigma = \Theta(1)$ (i.e., constant noise), a simple sampling strategy from the BO literature (Vakili et al. 2021) leads to $O^*\big(\lambda T^{-\frac{\nu}{2\nu+d}}\big)$ regret,[1] but the $\Omega(\lambda T^{-\frac{1}{2}})$ lower bound leaves open the possibility that a better algorithm might exist. On the other hand, at lower noise levels there are regimes where our upper and lower bounds match, as exemplified by the case $\sigma = \Theta(T^{-\frac{1}{4}})$ (and $\nu \geq \frac{d}{2}$) with regret $\Theta(\lambda T^{-\frac{3}{4}})$.

- Depending on the precise scaling of λ, the error bound can vary between those of BQ and BO. For instance, in the noiseless setting, we can control the rate of λ going to infinity as $\lambda = O(T^{\frac{\nu}{d}})$, and obtain a $\Omega(T^{-1})$ lower bound.

The connections to BQ and BO can partially be understood with the aid of the figures depicted in Figure 1. When λ is very small, estimating $\int_D e^{-\lambda f(\mathbf{x})} d\mathbf{x}$ is almost the same as estimating $\int_D (1 - \lambda f(\mathbf{x})) d\mathbf{x}$, which is a shifted and scaled version of $\int_D f(\mathbf{x}) d\mathbf{x}$. On the other hand, for large values of λ, the minimum value of $f(\mathbf{x})$ stands out considerably compared to the rest, as exemplified by the narrow bump in the figure.

A more detailed discussion of our contributions is deferred to the subsequent section on related work, where we provide a detailed analysis of the advancements and novel aspects of our research compared to existing works.

2 Related Work

Log-concave sampling. Over the years, a wide range of methods have been developed for estimating the normalizing constant of a probability distribution (Stoltz, Rousset et al. 2010), with a particular focus on algorithms that can leverage the log-concavity of the normalizing constant based on convex optimization (Lovász and Vempala 2006; Brosse, Durmus, and Moulines 2018; Dwivedi et al. 2018; Ge, Lee, and Lu 2020). These algorithms have been shown to have strong theoretical guarantees when provided with access to $\nabla f(\mathbf{x})$, by building on specific and popular bounds of Langevin-based sampling algorithms (Durmus and Moulines 2016).

Non-log-concave Sampling. Relatively fewer studies have explored the convergence rate for non-log-concave distributions, with the most related one being (Holzmüller and Bach 2023). Our results are generally not directly comparable with theirs due to the consideration of a different function class, leading to different algorithmic techniques (e.g., piecewise constant approximation vs. Gaussian process methods). In addition, one of our main goals is to handle noise, whereas (Holzmüller and Bach 2023) focuses on the noiseless setting. A more detailed comparison to (Holzmüller and Bach 2023) can be found in Appendix C.

Other Non-log-concave Sampling Works. Other works that consider non-convex energy functions appear to have significantly less direct relevance to our study due to the different modeling assumptions we adopt. However, it is worth mentioning a few of them, and interested readers can refer to the references provided in those papers for further exploration. Some of these works specifically focus on the analysis of *local* non-convexity within a small region, while the functions exhibit strong convexity outside of that region (Cheng et al. 2018; Ma et al. 2019). Additionally, there are studies that analyze upper and lower bounds related to the *relative Fisher information* by imposing restrictions on the smoothness of ∇f (Balasubramanian et al. 2022; Chewi et al. 2023).

Bayesian Optimization / GP Bandits. An important distinction that sets us apart from previous literature is that all of our results are rooted in the kernel-based bandit framework, which leverages Gaussian processes and Bayesian inference to approximate and optimize unknown functions. Within this area, two prominent problems are Bayesian quadrature (BQ) (Kanagawa and Hennig 2019; Wynne, Briol, and Girolami 2021; Cai, Lam, and Scarlett 2023) for approximating integrals and Bayesian optimization (BO) (Srinivas et al. 2010; Chowdhury and

[1] Monte-Carlo algorithms achieve $O(\frac{1}{\sqrt{T}})$ when $\lambda = \Theta(1)$, which is better than the mentioned $O^*\big(T^{-\frac{\nu}{2\nu+d}}\big)$ regret, but to our knowledge, its standard analysis is unable to capture the dependence on λ that might vary with T.

Gopalan 2017; Vakili et al. 2021) for finding the global maximum of a black-box function. These methods have gained popularity due to their wide range of applications and favorable theoretical properties. While BO is understood to be a more challenging problem than BQ in terms of the sample complexity, it can still be applied to provide sub-optimal BQ bounds (e.g., see (Cai, Lam, and Scarlett 2023)). In our case, it can similarly be utilized to obtain (sub-optimal) bounds for NC, thanks in part to the tight noisy L^∞ upper bound and upper bound on *information gain* derived by (Vakili et al. 2021; Vakili, Khezeli, and Picheny 2021).

We briefly note that various non-linear transformations of black-box functions (e.g., $e^{-\lambda f(x)}$ or $f(x)^2$) have appeared in other contexts such as optimization and regression (Flaxman, Teh, and Sejdinovic 2017; Astudillo and Frazier 2019; Marteau-Ferey, Bach, and Rudi 2020), but to our knowledge none have studied the NC problem.

3 Problem Setup

Let $f \in \mathcal{H}(B)$ be an RKHS function on the compact domain $D = [0,1]^d$,[2] where $\mathcal{H}(B)$ denotes the set of all functions whose RKHS norm $\|\cdot\|_{\mathcal{H}}$ is upper bounded by some constant $B > 0$. Here the RKHS norm $\|\cdot\|_{\mathcal{H}}$ depends on our choice of the kernel $k(\mathbf{x}, \mathbf{x}')$, and we focus our attention on the widely-adopted Matérn-ν kernel:

$$k(\mathbf{x}, \mathbf{x}') = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}\,\|\mathbf{x} - \mathbf{x}'\|}{l} \right)^{\nu} J_\nu \left(\frac{\sqrt{2\nu}\,\|\mathbf{x} - \mathbf{x}'\|}{l} \right),$$

where J_ν is the modified Bessel function, and Γ is the gamma function. To make the dependence on ν explicit, we use $\mathcal{H}^\nu$ to denote the Matérn RKHS, with its RKHS norm being represented by $\|\cdot\|_{\mathcal{H}^\nu}$. Given query access to $f(\mathbf{x})$, at time step t, the observations y_t are modeled as follows:

- In the noiseless setting, we simply have $y_t = f(\mathbf{x}_t)$.
- In the noisy setting, we have $y_t = f(\mathbf{x}_t) + z_t$, where $z_t \sim \mathcal{N}(0, \sigma^2)$ is i.i.d. Gaussian noise.

Our goal is to approximate the normalizing constant,

$$Z(f) = \int_D e^{-\lambda f(\mathbf{x})} d\mathbf{x}, \quad \lambda > 0, \tag{1}$$

leading to an estimate $\hat{Z}(f)$, which we seek to be accurate within a multiplicative factor of $1 \pm \epsilon$. In other words, we are interested in the quantity

$$\epsilon = \sup_{f \in \mathcal{H}^\nu} \left| \frac{\hat{Z}(f)}{Z(f)} - 1 \right|. \tag{2}$$

While the additive error (which would be $|\hat{Z} - Z|$ in our case) is commonly used for integration problems such as BQ, it is less suitable here unless $\lambda \to 0$ (yielding $Z \to 1$). This is because as λ grows large, Z may approach zero (in which case $|\hat{Z} - Z| \le \epsilon$ may hold trivially) or infinity (in which case $|\hat{Z} - Z| \le \epsilon$ may be an overly stringent requirement). The relative error (2) serves to more naturally unify these various cases. In a similar manner, (Holzmüller and Bach 2023) defines the error as $\epsilon = \left| \log \hat{Z} - \log Z \right|$, which is essentially equivalent to (2) due to the fact that $\log(1+\alpha)$ behaves as $O(\alpha)$ when $|\alpha|$ is strictly smaller than one (and as $\alpha(1 + o(1))$ when $\alpha \to 0$).

[2] Any rectangular domain can be reduced to $[0,1]^d$ by suitable shifting and scaling.

4 Lower Bounds

In this section, we provide algorithm-independent lower bounds on ϵ (see (2)), i.e., impossibility/hardness results, for both the noiseless and noisy settings.

Theorem 1. (Noiseless Lower Bound) *Consider the noiseless setting with $f \in \mathcal{H}^\nu(B)$, $\nu + \frac{d}{2} \ge 1$, and a time horizon $T \to \infty$. For any algorithm that estimates Z and produces an estimate $\hat{Z}$ satisfying* (2)*, the worst-case $f \in \mathcal{H}^\nu(B)$ must have the following lower bound on* (2) *with $\Omega(1)$ probability:*

- *If $\lambda = \Theta(T^c)$ with $c \le \frac{\nu}{d} + \frac{1}{2}$, then $\epsilon = \Omega(T^{-\frac{\nu}{d} - 1 + c})$.*
- *If $\lambda = \Theta(\log T)$, then $\epsilon = \Omega(T^{-\frac{\nu}{d} - 1} \log T)$.*

The condition $c \le \frac{\nu}{d} + \frac{1}{2}$ comes from a technical condition in the proof, and we believe it to be quite mild (and similarly in other results below); in particular, we cover broad cases of interest, some of which are given as follows:

- For $c = 0$, we have when $\lambda = \Theta(1)$, $\epsilon = \Omega(T^{-\frac{\nu}{d} - 1})$.
- For $c = \frac{1}{2}$, we have when $\lambda = \Theta(\sqrt{T})$, $\epsilon = \Omega(T^{-\frac{\nu}{d} - \frac{1}{2}})$.
- For $c = \frac{\nu}{d}$, we have when $\lambda = \Theta(T^{\frac{\nu}{d}})$, $\epsilon = \Omega(T^{-1})$.
- For $c = \frac{\nu}{d} + \frac{1}{2}$, when $\lambda = \Theta(T^{\frac{\nu}{d} + \frac{1}{2}})$, $\epsilon = \Omega(T^{-\frac{1}{2}})$.
- For $c = 1$ (which requires $\nu \ge \frac{d}{2}$), we have when $\lambda = \Theta(T)$, $\epsilon = \Omega(T^{-\frac{\nu}{d}})$.

Note that the noiseless BQ lower bound is known as $\epsilon = \Omega(T^{-\frac{\nu}{d} - 1})$ (see e.g. (Novak 2006; Cai, Lam, and Scarlett 2023)), and the noiseless BO lower bound is $\epsilon = \Omega(T^{-\frac{\nu}{d}})$ (Bull 2011). Thus, the above results indicate that when $\lambda = \Theta(1)$ or approaches 0, the lower bound matches that of BQ, whereas when $\lambda = \Theta(T)$, the bound matches that of BO. Although the above arguments compare additive error with relative error, this is valid since our hard function class in proving the lower bound has $\Theta(1)$ normalizing constant. See Appendix A for further details.

Next, we turn our attention to the noisy setting.

Theorem 2. (Noisy Lower Bound) *Consider the noisy setting with $f \in \mathcal{H}^\nu(B)$, $\nu + \frac{d}{2} \ge 1$, a time horizon $T \to \infty$, and noise variance σ^2 (possibly varying with T). For any algorithm that estimates Z and produces an estimate $\hat{Z}$ satisfying* (2)*, the worst-case $f \in \mathcal{H}^\nu(B)$ must have the following lower bound on* (2) *with $\Omega(1)$ probability:*

- *If $\lambda = \Theta(T^c)$ and $\sigma = \Theta(T^a)$ with $c \le \min\left\{ \frac{\nu}{d} + \frac{1}{2}, (\frac{1}{2} - a)\frac{2\nu + d}{2\nu + 2d} \right\}$ and $a \le \frac{1}{2}$, then $\epsilon = \Omega(T^{-\frac{\nu}{d} - 1 + c} + \sigma T^{-\frac{1}{2} + c})$.*
- *If $\lambda = \Theta(\log T)$ and $\sigma = \Theta(T^a)$ with $a < \frac{1}{2}$, then $\epsilon = \Omega(T^{-\frac{\nu}{d} - 1} \log T + \sigma T^{-\frac{1}{2}} \log T)$.*

Since the noisy setting involves the variables a and c, we will focus mainly on the most studied case where $\sigma = \Theta(1)$ ($a = 0$) for the sake of clarity. For constant noise

variance, the BQ noisy lower bound is known to be $\Omega(T^{-\frac{1}{2}})$ (Plaskota 1996; Cai, Lam, and Scarlett 2023), and the BO noisy lower bound is known as $\epsilon = \Omega(T^{-\frac{\nu}{2\nu+d}})$ (Scarlett, Bogunovic, and Cevher 2017; Cai and Scarlett 2021). To compare against these, for the first dot point in Theorem 2, we consider the following specific c values:

- For $c = 0$, we have when $\lambda = \Theta(1)$, $\epsilon = \Omega(T^{-\frac{1}{2}})$, which coincides with the noisy BQ lower bound.
- For $c = \frac{d}{4\nu+2d}$ (needs $\nu \geq \frac{\sqrt{5}-1}{4}d$, so that $\frac{d}{4\nu+2d} \leq \frac{2\nu+d}{4\nu+4d}$), we have when $\lambda = \Theta(T^{\frac{d}{4\nu+2d}})$, $\epsilon = \Omega(T^{-\frac{\nu}{2\nu+d}})$, which coincides with the noisy BO lower bound. Note that this particular choice of λ is somewhat artificial, but it highlights the fact that our result can lead to BO-like results.
- For $c = \frac{1}{3}$ (needs $\nu \geq \frac{d}{2}$), we have when $\lambda = \Theta(T^{\frac{1}{3}})$, $\epsilon = \Omega(T^{-\frac{1}{6}})$.
- For $c = \frac{1}{4}$, we have when $\lambda = \Theta(T^{\frac{1}{4}})$, $\epsilon = \Omega(T^{-\frac{1}{4}})$.
- For $c = \frac{2\nu+d}{4\nu+4d}$ (the maximum allowed value), we have when $\lambda = \Theta(T^{\frac{2\nu+d}{4\nu+4d}})$, $\epsilon = \Omega(T^{-\frac{d}{4\nu+4d}})$.

Further analysis of this lower bound can be found in Appendix A; similar to the above results, we may end at BQ or BO type of error bounds for different a and c values. To illustrate an example, we recall the result shown in Table 1 when $a = -\frac{1}{2}$, where the lower bound is $\Omega(\lambda T^{-\frac{1}{2}})$. Substituting $a = -\frac{1}{2}$ into Theorem 2, it can be seen that the value of c lies in the range $(-\infty, \frac{2\nu+d}{2\nu+2d}]$. By choosing $\lambda = \Theta(T^{-\frac{\nu}{d}})$, we obtain the BQ-like lower bound $\Omega = \Omega(T^{-\frac{\nu}{d}-\frac{1}{2}})$, whereas by choosing $\lambda = \Theta(T^{\frac{2\nu+d}{2\nu+2d}})$, we obtain the BO-like lower bound $\Omega = \Omega(T^{-\frac{d}{4\nu+4d}})$.

In the following section, we will derive algorithmic upper bounds that sometimes match the algorithm-indpeendent lower bounds, though with gaps remaining in other cases.

5 Upper Bounds

We present a GP-based two-batch algorithm for estimating the normalizing constant in Algorithm 1. Given a GP prior model $\mathrm{GP}(0, k)$, after observing t samples, the posterior distribution is also a GP with the following posterior mean and variance:

$$\mu_t(\mathbf{x}) = \mathbf{k}_t(\mathbf{x})^T \big(\mathbf{K}_t + \xi \mathbf{I}_t\big)^{-1} \mathbf{y}_t, \tag{3}$$

$$\sigma_t^2(\mathbf{x}) = k(\mathbf{x}, \mathbf{x}) - \mathbf{k}_t(\mathbf{x})^T \big(\mathbf{K}_t + \xi \mathbf{I}_t\big)^{-1} \mathbf{k}_t(\mathbf{x}), \tag{4}$$

where $\mathbf{y}_t = [y_1, \ldots, y_t]^T$, $\mathbf{k}_t(\mathbf{x}) = \big[k(\mathbf{x}_i, \mathbf{x})\big]_{i=1}^t$, $\mathbf{K}_t = \big[k(\mathbf{x}_t, \mathbf{x}_{t'})\big]_{t,t'}$ is the kernel matrix, $\mathbf{I}_t$ is the identity matrix of dimension t, and $\xi > 0$ is a hyperparameter.

In Algorithm 1, we initially employ $\frac{T}{2}$ samples to construct a GP approximation of f in a non-adaptive manner (i.e., the selection of $\mathbf{x}_1, \ldots, \mathbf{x}_{T/2}$ occurs prior to observing $y_1, \ldots, y_{T/2}$). This gives us an estimate $\mu_{T/2}(\cdot)$ of the entire function, from which we can form an initial estimate $\hat{Z}_1$ of Z. We then refine the estimate by using Monte Carlo sampling to estimate a multiplicative "residual" term (with estimate $\hat{R}$), and the final estimate $\hat{Z}$ is the product $\hat{Z}_1 \cdot \hat{R}$.

Algorithm 1: Two-batch normalizing constant estimation algorithm

Input: Domain D, $\mathrm{GP}(0, k)$ prior, GP hyperparameter ξ, time horizon T, noise standard deviation σ.
for $t = 1, \ldots, T/2$ **do**
  Select $\mathbf{x}_t = \arg\max_{\mathbf{x} \in D} \sigma_{t-1}(\mathbf{x})$.
  Update σ_t using $\mathbf{x}_1, \ldots, \mathbf{x}_t$.
end for
Update $\mu_{T/2}(\mathbf{x})$ using $\{\mathbf{x}_t\}_{t=1}^{T/2}$ and $\{y_t\}_{t=1}^{T/2}$.
for $t = T/2 + 1, \ldots, T$ **do**
  Sample $\mathbf{x}_t \sim \frac{e^{-\lambda \mu_{T/2}(\mathbf{x})}}{\int_D e^{-\lambda \mu_{T/2}(\mathbf{x})} d\mathbf{x}}$ independently.
  Receive $y_t = f(\mathbf{x}_t) + z_t$
end for
Compute the approximate integral $\hat{Z}_1 = \int_D e^{-\lambda \mu_{T/2}(\mathbf{x})} d\mathbf{x}$
Compute the residual $\hat{R} = \frac{2}{T e^{\lambda^2 \sigma^2/2}} \sum_{t=T/2+1}^{T} e^{\lambda \mu_{T/2}(\mathbf{x}_t) - \lambda y_t}$
Output $\hat{Z} = \hat{Z}_1 \cdot \hat{R}$

The two-batch idea is most easily understood in the BQ problem of integrating f: Roughly speaking, the error of Monte Carlo decays as $\frac{1}{\sqrt{T}}$ but is also proportional to the function scale itself, so by applying it to the residual, we get the original error times $\frac{1}{\sqrt{T}}$. The intuition in our NC problem is generally similar, but the non-linearity of $e^{-\lambda f}$ leads us to consider a multiplicative residual, and the analysis becomes more complicated.

As discussed in Section 2, (Holzmüller and Bach 2023) has utilized a similar algorithm by selecting grid points and forming a piecewise constant approximation in the first batch, which they find to be optimal for a different once-differentiable function class, but is less suitable for our setting (see Appendix C and Section 6). Simpler variants of this idea have also been used in BQ; see (Cai, Lam, and Scarlett 2023) and the references therein.

We summarize the error bounds of Algorithm 1 (as well as the estimate resulting from the first batch alone) in the following two theorems.

Theorem 3. (Noiseless Upper Bound) *Consider our problem setup with constant parameters (B, ν, d, l), and time horizon $T \to \infty$. With probability at least $1 - \delta$ (for an arbitrary fixed $\delta \in (0, 1)$), the relative error incurred by Algorithm 1 with $\xi = 0$ has the following upper bounds on* (2)*:*

- *If $\lambda = \Theta(T^c)$ with $c \leq \frac{\nu}{d}$, then $\epsilon = O\big(T^{-\frac{\nu}{d}-\frac{1}{2}+c}\big)$.*
- *If $\lambda = \Theta(\log T)$, then $\epsilon = O\big(T^{-\frac{\nu}{d}-\frac{1}{2}} \log T\big)$.*

Comparing to the noiseless lower bound obtained in Theorem 1, there is a non-negligible gap of $O(\sqrt{T})$. However, this difference is fairly insignificant when $\nu \gg d$.

Theorem 4. (Noisy Upper Bound) *Consider our problem setup with constant parameters (B, ν, d, l, ξ), noise*

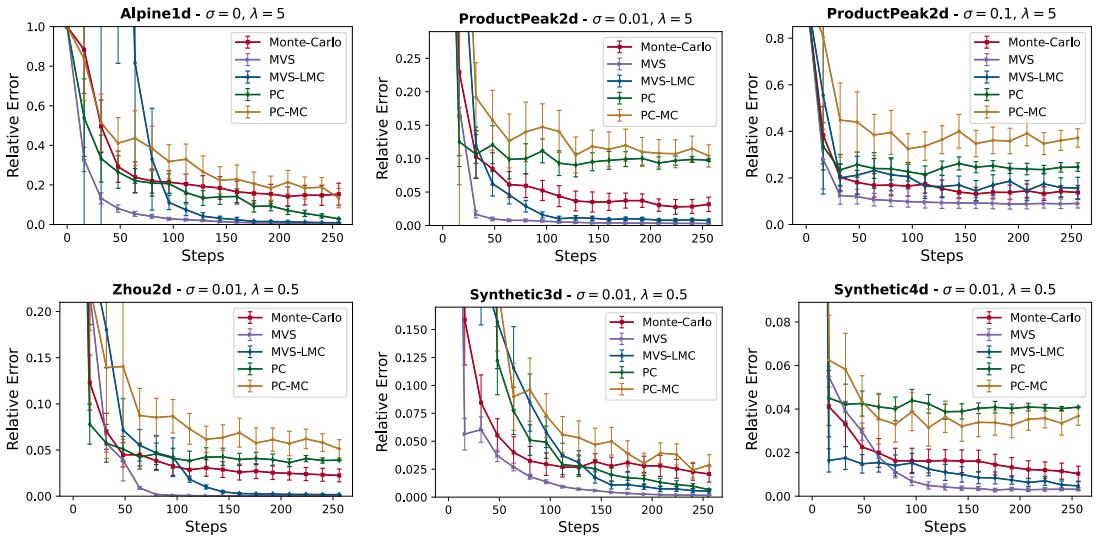


Figure 2: Results for analytic functions.

standard deviation σ *that may scale with* T *(i.e.,* $\sigma = \Theta(T^a)$*), and time horizon* $T \to \infty$*. Then, the following upper bounds on* (2) *hold with probability at least* $1 - \delta$ *for any constant* $\delta > 0$*:*

- *If* $\lambda = \Theta(T^c)$ *with* $a + c \leq 0$ *and* $c < \frac{\nu}{2\nu+d}$*, Algorithm 1 yields* $\epsilon = O\big(T^{-\frac{\nu}{2\nu+d}-\frac{1}{2}+c}(\log T)^{\frac{\nu}{2\nu+d}} + \sigma T^{-\frac{1}{2}+c}\big)$.
- *If* $\lambda = \Theta(\log T)$ *with* $a \leq 0$*, Algorithm 1 yields* $\epsilon = O\big(T^{-\frac{\nu}{2\nu+d}-\frac{1}{2}}(\log T)^{\frac{3\nu+d}{2\nu+d}} + \sigma T^{-\frac{1}{2}} \log T\big)$.

In addition, the following upper bounds hold with probability at least $1 - \frac{1}{T^\alpha}$ *with* α *being any fixed constant:*

- *If* $\lambda = \Theta(T^c)$ *with* $a + c < \frac{\nu}{2\nu+d}$ *and* $c < \frac{\nu}{2\nu+d}$*, the intermediate estimate* $\hat{Z}_1$ *from Algorithm 1 yields* $\epsilon = O\big(T^{-\frac{\nu}{2\nu+d}+c}(\log T)^{\frac{\nu}{2\nu+d}} + \sigma T^{-\frac{\nu}{2\nu+d}+c}(\log T)^{\frac{4\nu+d}{4\nu+2d}}\big)$.
- *If* $\lambda = \Theta(\log T)$ *with* $a < \frac{\nu}{2\nu+d}$*, the intermediate estimate* $\hat{Z}_1$ *from Algorithm 1 yields* $\epsilon = O\big(T^{-\frac{\nu}{2\nu+d}}(\log T)^{\frac{3\nu+d}{2\nu+d}} + \sigma T^{-\frac{\nu}{2\nu+d}}(\log T)^{\frac{8\nu+3d}{4\nu+2d}}\big)$.

The second part of the theorem helps to broaden the range of allowed (a, c) pairs. However, when (a, c) is feasible for the first part, it gives a stronger result than the second part, improving by a $O(\frac{1}{\sqrt{T}})$ factor in the non-σ term and by $O(\frac{d}{4\nu+2d})$ in the σ term. This highlights the benefit of using both batches, at least in terms of upper bounds.

Comparing to the noisy lower bound obtained in Theorem 2, the convergence rate obtained in Theorem 4 is less straightforward, but we give some analysis as follows:

- For the first dot points in Theorem 2 and Theorem 4, the two results align under the high noise regime when $a \geq -\frac{\nu}{2\nu+d}$ (this threshold on a is obtained by equating the exponents $-\frac{\nu}{2\nu+d} - \frac{1}{2} + c$ and $a - \frac{1}{2} + c$ from Theorem 4). In this regime, the order of the error is optimal as $\Theta(\sigma T^{-\frac{1}{2}+c})$. This matches the order-optimal bound for BQ (upon replacing c by zero, since λ is absent in BQ). As a specific example, as shown in Table 1, when we assume $a = -\frac{1}{4}$ and $\nu \geq \frac{d}{2}$ (which ensures $-\frac{1}{4} \geq -\frac{\nu}{2\nu+d}$), the resulting upper bound is optimal at $\Theta(T^{-\frac{3}{4}+c})$.
- For an extreme low noise regime ($a \leq -\frac{\nu}{d} - \frac{1}{2}$, which ensures that the first term in Theorem 2 dominates the second term), the lower bound vs. the upper bound is $\Omega(T^{-\frac{\nu}{d}-1+c})$ vs. $O(T^{-\frac{\nu}{2\nu+d}-\frac{1}{2}+c})$. Thus, the relative gap is $O(T^{\frac{d}{4\nu+2d}+\frac{\nu}{d}})$, which is around $O(\sqrt{T})$ if $d \gg \nu$.
- Otherwise, if $-\frac{\nu}{d} - \frac{1}{2} < a < -\frac{\nu}{2\nu+d}$, the lower bound vs. the upper bound is $\Omega(\sigma T^{-\frac{1}{2}+c})$ vs. $O(T^{-\frac{\nu}{2\nu+d}-\frac{1}{2}+c})$. The relative gap is now $O(T^{-\frac{\nu}{2\nu+d}-a})$, which primarily depends on the value of a if $d \gg \nu$. In this sense, the gap ranges from $O(T^{o(1)})$ to $O(T)$. To illustrate an example under this condition, see Table 1 with the choice $a = -\frac{1}{2}$.
- The last two bullet points in the theorem address the remaining scenario when $\lambda\sigma \to \infty$ (i.e., $a + c > 0$), and the term $\frac{\nu}{2\nu+d}$ in the exponent matches that observed for simple regret in the BO literature (Scarlett, Bogunovic, and Cevher 2017; Vakili et al. 2021), though failing to match the upper bound for NC. Hence, the aforementioned noisy upper bounds demonstrate that, albeit with some gaps, when $\lambda\sigma \to 0$, the derived upper bound shares similarities with BQ, whereas when $\lambda\sigma \to \infty$, the bound shares similarities with BO.

6 Experiments

In this section, we conduct simulation studies to investigate Algorithm 1 and its intermediate estimate $\hat{Z}_1$.

Setup

Sampling in the Second Batch. As shown in line 8 of Algorithm 1, the target distribution is proportional to

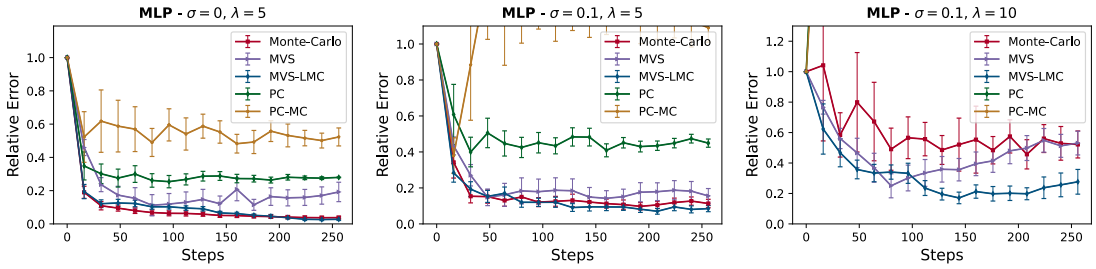


Figure 3: Results for MLP.

$e^{-\lambda\mu_{T/2}(\mathbf{x})}$, which can be challenging to sample exactly. Fortunately, there is a vast literature on approximate sampling methods that we can use. We choose to use the method of Langevin dynamics defined by the following stochastic differential equation (SDE) (Uhlenbeck and Ornstein 1930):

$$d\mathbf{X}(t) = -\nabla g(\mathbf{X}(t))dt + \sqrt{2\lambda^{-1}}\mathbf{B}(t), \tag{5}$$

where $\lambda > 0$ is interpreted as the inverse temperature, and $\mathbf{B}(t) \in \mathbb{R}^d$ is the Brownian motion at time t. A standard approach to solve (5) is to apply Euler-Maruyama discretization, leading to the following *Langevin Monte-Carlo* (LMC) updating rule:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \beta\nabla\mu_{T/2}(\mathbf{x}_t) + \sqrt{2\beta\lambda^{-1}}\boldsymbol{\epsilon}_t, \tag{6}$$

where we have replaced g with $\mu_{T/2}$, $\boldsymbol{\epsilon}_t$ are i.i.d. standard Gaussian random vectors in $\mathbb{R}^d$, and $\beta > 0$ is the step size of the discretization. Note that using this sampling strategy incurs some approximation error that we do not attempt to account for in our theory (analogous to how BO theory assumes exact acquisition function optimization).

Hyperparameters. For all functions considered in this section, we consider a time horizon of $T = 256$, $\lambda \in \{0.5, 5, 10\}$, $\sigma \in \{0, 0.01, 0.1\}$ and $\nu \in \{0.5, 1.5, 2.5\}$. The total number of steps of (6) is set as 20, and the LMC learning rate is $\beta = 10^{-3}$. We adopt two learnable kernel hyperparameters in (3), the lengthscale l and an additional scale parameter (multiplying k), to permit functions with varying ranges (while ν remains fixed). Except for synthetic functions where the true hyperparameters are known, these two parameters are optimized by maximizing the data log-likelihood (Rasmussen and Williams 2006) using the built-in SciPy optimizer based on L-BFGS-B, which is also used for finding the maximum variance point in Algorithm 1.

Benchmarks. In addition to the commonly adopted Monte-Carlo quadrature baseline, as discussed in Sections 2 and 5, the most closely related work by (Holzmüller and Bach 2023, Sec. 5.1) proposes the use of piecewise constant approximation to estimate NC with grid inputs, which also achieves improved theoretical convergence when combined with an additional MC step. We adopt their shorthand notations and refer to these two benchmarks as PC and PC-MC, respectively.

Evaluation. We refer to the first batch of Algorithm 1 as maximum variance sampling (MVS),[3] and the whole Algorithm 1 as MVS-LMC. We evaluate the performance using the mean absolute relative error, with the ground truth value (and also $\hat{Z}_1$ at Line 11 of Algorithm 1) being determined by trapezoidal rule with 10^5 uniformly-spaced grid points (without noise). Error bars in our plots indicate ± 0.5 standard deviation with respect to the 100 trials.

Analytic Functions

In order to assess the empirical behaviour of MVS and MVS-LMC, we first conduct experiments on the following analytic functions for $d \in \{1, 2, 3, 4\}$:

Synthetic functions. The synthetic functions are constructed by sampling $m = 30d$ points, $\hat{\mathbf{x}}_1 \dots \hat{\mathbf{x}}_m$, uniformly on $[0, 1]^d$, and $\hat{a}_1 \dots \hat{a}_m$ uniformly on $[-1, 1]$. The function is then defined as $f(\mathbf{x}) = \sum_{i=1}^{m} \hat{a}_i k(\hat{\mathbf{x}}_i, \mathbf{x})$. The length-scale and ν are set to be fixed (no hyperparameter learning) as 0.2 and 2.5 respectively.

Benchmark functions. Exact formulations of functions including, Ackley, Alpine, Product-Peak, Zhou, etc., can be found in (Bingham 2013). ν is fixed as $3/2$ for all of these benchmark functions.

Multi-layer Perception (MLP)

For a more complex scenario, we consider an 8-dimensional MLP function, with the structure being defined by

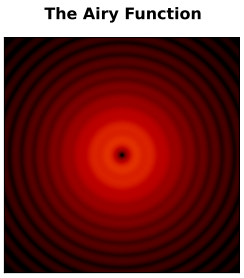
$$f(\mathbf{x}) = f^4(\mathrm{Tanh}(f^3(\mathrm{Tanh}(f^2(\mathrm{Tanh}(f^1(\mathbf{x})))))))),$$

where $\mathrm{Tanh}(\cdot) : \mathbb{R}^n \to [-1, 1]^n$ is the Hyperbolic tangent (tanh) activation function, and the dimension mapping from layers f^1 to f^4 is $8 \to 16 \to 32 \to 16 \to 1$. We use Xavier initialization to set and fix the weights of the MLP, and set $\nu = 1/2$ to model the potentially more erratic behavior.

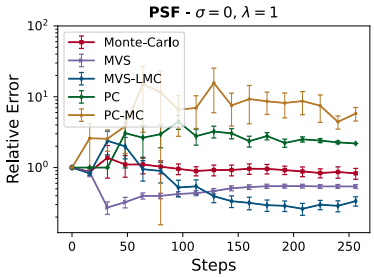
Point Spread Function (PSF)

Beyond functions with analytic forms, we have also simulate on a diffraction energy distribution characterised by an intensity of wave-field (i.e., PSF). This idea leads to an interesting class of functions for black-box problems, e.g., it has been previously evaluated in BQ tasks (Naslidnyk, Gonzalez, and Mahsereci 2021). The PSF will be dependent

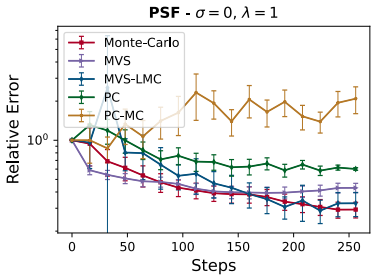
[3] More precisely, MVS corresponds to taking all T samples based on the maximum variance rule, not just the first $T/2$.



(a) PSF computed for the wavelength 2×10^{-6}.



(b) PSF error plot.



(c) PSF error plot by shifting $(+0.05, +0.05)$, so that the new optimum is at $(-0.05, -0.05)$.

Figure 4: Result for estimating PSF.

on the shape of the pupil (circle, rectangle, etc.), and on the wavelength of the used light. In the case of a circular pupil, the diffracted wave pattern is known as the Airy pattern, where in our case, the logarithm of the energy intensity is regarded as a 2-D black-box energy function. The Airy pattern generated by light with wavelength 2×10^{-6} is displayed in Figure 4a, and we perform NC estimation within a quarter of the support, namely $[0, 0.5] \times [0, 0.5]$ (top-right corner of Figure 4a). Additionally, Figure 4b displays the error curves (plotted with $\nu = 0.5$, $\sigma = 0$, and $\lambda = 1$) using a logarithmic scale to capture the behavior of all the algorithms more effectively.

Discussion

As can be observed from the displayed figures, the performance of MVS and MVS-LMC outperforms MC, PC, and PC-MC by varying margins. We find that despite the PC approach working well for higher query budgets (Holzmüller and Bach 2023) (e.g., 10^5 or more), GP-based methods tend to be superior at low budgets (e.g., a few hundred). This is in alignment with the strong query complexity properties of GP methods observed in other tasks (e.g., regression, optimization, etc.).

In particular, the number of dimensions can significantly impact the performance of PC under a small budget, as observations become more sparse in higher dimensions. For example, in the Alpine1d plot, PC converges quickly, while it converges slowly for Synthetic4d. On the other hand, in the PSF experiment, PC suffers from a relative error around 3 due to consistently sampling from the $(0, 0)$ point, which has a significantly higher value than other locations. As seen in Figure 4c, this can be improved by shifting f so that the peak is away from $(0, 0)$, but only to a limited extent.

When comparing MVS and MVS-LMC to MC on simple analytic functions (see Figure 2), the empirical behavior is consistent with our theory; that is, for a small inverse temperature λ and noise variance σ^2, both MVS and MVS-LMC exhibit a substantial improvement over MC. However, for larger values of λ and σ, the error of MVS-LMC essentially reduces to $O(T^{-\frac{1}{2}})$, the same order as MC.

We observe that MVS-LMC works particularly well for the MLP and PSF functions, whose results are shown in Figures 3 and 4b. While the total samples are split half-by-half in Algorithm 1 for simplicity, the performance of MVS-LMC could be further improved in practice by choosing a problem-dependent split size, similarly to BQ in (Cai, Lam, and Scarlett 2023). We also note that MVS usually also works well, indicating that the LMC component is not always necessary, but the LMC component clearly helps in some cases (specifically, for MLP and PSF).

Effect of Varying λ

Our theoretical results indicate that the complexity of NC increases with higher values of λ. This trend is also observed empirically, as demonstrated by the MLP plots in Figure 3. See also Appendix D for additional results of this kind.

7 Conclusion

Our work contributes to the understanding of the estimation of the normalizing constant for functions in an RKHS, and provides insights into the relationship between the error bound, the problem parameter λ, and the noise variance σ^2. In general it is still an open question to what extent our bounds can be improved. Our upper bounds on the convergence were primarily established using L^∞ bounds in BO, which leads to an extra $\Theta(\sqrt{T})$ factor compared to our lower bounds. Improvements may be possible if we can instead build on L^2 function approximation bounds. It would also be of interest to better understand the *squared exponential* (SE) kernel, for which some of our techniques become infeasible (e.g., the use of disjoint bump functions in the lower bound).

Acknowledgments

This work was supported by the Singapore National Research Foundation (NRF) under grant number A-0008064-00-00.

References

Astudillo, R.; and Frazier, P. 2019. Bayesian optimization of composite functions. In *International Conference on Machine Learning*.

Bakhvalov, N. S. 1959. On the approximate calculation of multiple integrals. *Vestnik MGU, Ser. Math. Mech. Astron. Phys. Chem*, 4: 3–18.

Balasubramanian, K.; Chewi, S.; Erdogdu, M. A.; Salim, A.; and Zhang, S. 2022. Towards a theory of non-log-concave sampling: first-order stationarity guarantees for langevin monte carlo. In *Conference on Learning Theory*, 2896–2923. PMLR.

Bingham, D. 2013. Virtual Library of Simulation Experiments: Test Functions and Datasets. https://www.sfu.ca/~ssurjano/index.html. Accessed: 2023-08-05.

Boucheron, S.; Lugosi, G.; and Bousquet, O. 2004. Concentration inequalities. In *Summer school on machine learning*, 208–240. Springer.

Brosse, N.; Durmus, A.; and Moulines, É. 2018. Normalizing constants of log-concave densities. *Electronic Journal of Statistics*, 12(1): 851 – 889.

Bull, A. D. 2011. Convergence rates of efficient global optimization algorithms. *Journal of Machine Learning Research*, 12(10).

Cai, X.; Lam, T.; and Scarlett, J. 2023. On average-case error bounds for kernel-based Bayesian quadrature. *Transactions on Machine Learning Research*.

Cai, X.; and Scarlett, J. 2021. On lower bounds for standard and robust Gaussian process bandit optimization. In *International Conference on Machine Learning*, 1216–1226. PMLR.

Chen, M.-H.; and Shao, Q.-M. 1997. On Monte Carlo methods for estimating ratios of normalizing constants. *The Annals of Statistics*, 25(4): 1563–1594.

Cheng, X.; Chatterji, N. S.; Abbasi-Yadkori, Y.; Bartlett, P. L.; and Jordan, M. I. 2018. Sharp convergence rates for Langevin dynamics in the nonconvex setting. *arXiv preprint arXiv:1805.01648*.

Chewi, S.; Gerber, P.; Lee, H.; and Lu, C. 2023. Fisher information lower bounds for sampling. In *International Conference on Algorithmic Learning Theory*, 375–410. PMLR.

Chowdhury, S. R.; and Gopalan, A. 2017. On kernelized multi-armed bandits. In *International Conference on Machine Learning*, 844–853. PMLR.

Desjardins, G.; Bengio, Y.; and Courville, A. C. 2011. On tracking the partition function. *Advances in Neural Information Processing Systems*, 24.

Durmus, A.; and Moulines, É. 2016. High-dimensional Bayesian inference via the unadjusted Langevin algorithm. *Bernoulli*.

Dwivedi, R.; Chen, Y.; Wainwright, M. J.; and Yu, B. 2018. Log-concave sampling: Metropolis-Hastings algorithms are fast! In *Conference on learning theory*, 793–797. PMLR.

Flaxman, S.; Teh, Y. W.; and Sejdinovic, D. 2017. Poisson intensity estimation with reproducing kernels. In *International Conference on Artificial Intelligence and Statistics*, 270–279. PMLR.

Ge, R.; Lee, H.; and Lu, J. 2020. Estimating normalizing constants for log-concave distributions: Algorithms and lower bounds. In *ACM SIGACT Symposium on Theory of Computing*, 579–586.

Gelman, A.; and Meng, X.-L. 1998. Simulating normalizing constants: From importance sampling to bridge sampling to path sampling. *Statistical science*, 163–185.

Holzmüller, D.; and Bach, F. 2023. Convergence rates for non-log-concave sampling and log-partition estimation. *arXiv preprint arXiv:2303.03237*.

Kanagawa, M.; and Hennig, P. 2019. Convergence guarantees for adaptive Bayesian quadrature methods. *Advances in Neural Information Processing Systems*, 32.

Kanagawa, M.; Hennig, P.; Sejdinovic, D.; and Sriperumbudur, B. K. 2018. Gaussian processes and kernel methods: A review on connections and equivalences. *arXiv preprint arXiv:1807.02582*.

Lovász, L.; and Vempala, S. 2006. Fast algorithms for logconcave functions: Sampling, rounding, integration and optimization. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, 57–68. IEEE.

Ma, Y.-A.; Chen, Y.; Jin, C.; Flammarion, N.; and Jordan, M. I. 2019. Sampling can be faster than optimization. *Proceedings of the National Academy of Sciences*, 116(42): 20881–20885.

Marteau-Ferey, U.; Bach, F.; and Rudi, A. 2020. Non-parametric models for non-negative functions. *Advances in Neural Information Processing Systems*, 33: 12816–12826.

Naslidnyk, M.; Gonzalez, J.; and Mahsereci, M. 2021. Invariant priors for Bayesian quadrature. *arXiv preprint arXiv:2112.01578*.

Novak, E. 2006. *Deterministic and stochastic error bounds in numerical analysis*, volume 1349. Springer.

Oates, C. J.; Girolami, M.; and Chopin, N. 2017. Control functionals for Monte Carlo integration. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 695–718.

Plaskota, L. 1996. Worst case complexity of problems with random information noise. *Journal of Complexity*, 12(4): 416–439.

Rasmussen, C. E.; and Williams, C. K. I. 2006. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press.

Ripley, B. D. 2009. *Stochastic simulation*. John Wiley & Sons.

Scarlett, J.; Bogunovic, I.; and Cevher, V. 2017. Lower bounds on regret for noisy Gaussian process bandit optimization. In *Conference on Learning Theory*, 1723–1742. PMLR.

Srinivas, N.; Krause, A.; Kakade, S. M.; and Seeger, M. 2010. Gaussian process optimization in the bandit

setting: no regret and experimental design. In *International Conference on Machine Learning*.

Stoltz, G.; Rousset, M.; et al. 2010. *Free energy computations: A mathematical perspective*. World Scientific.

Uhlenbeck, G. E.; and Ornstein, L. S. 1930. On the theory of the Brownian motion. *Physical Review*, 36(5): 823.

Vakili, S.; Bouziani, N.; Jalali, S.; Bernacchia, A.; and Shiu, D.-s. 2021. Optimal order simple regret for Gaussian process bandits. *Advances in Neural Information Processing Systems*, 34: 21202–21215.

Vakili, S.; Khezeli, K.; and Picheny, V. 2021. On information gain and regret bounds in Gaussian process bandits. In *International Conference on Artificial Intelligence and Statistics*, 82–90. PMLR.

Wendland, H.; and Rieger, C. 2005. Approximate interpolation with applications to selecting smoothing parameters. *Numerische Mathematik*, 101(4): 729–748.

Wynne, G.; Briol, F.-X.; and Girolami, M. 2021. Convergence guarantees for Gaussian process means with misspecified likelihoods and smoothness. *Journal of Machine Learning Research*, 22.