Statistics and Computing (2020) 30:839–853
https://doi.org/10.1007/s11222-020-09922-0

Efficient Bayesian shape-restricted function estimation with constrained Gaussian process priors

Pallavi Ray[1] · **Debdeep Pati**[1] · **Anirban Bhattacharya**[1]

Received: 7 February 2019 / Accepted: 7 January 2020 / Published online: 30 January 2020

Abstract
This article revisits the problem of Bayesian shape-restricted inference in the light of a recently developed approximate Gaussian process that admits an equivalent formulation of the shape constraints in terms of the basis coefficients. We propose a strategy to efficiently sample from the resulting constrained posterior by absorbing a *smooth relaxation* of the constraint in the likelihood and using circulant embedding techniques to sample from the unconstrained *modified prior*. We additionally pay careful attention to mitigate the computational complexity arising from updating hyperparameters within the covariance kernel of the Gaussian process. The developed algorithm is shown to be accurate and highly efficient in simulated and real data examples.

Keywords Circulant embedding · Durbin's recursion · Elliptical slice sampling · Smooth relaxation · Toeplitz

1 Introduction

In diverse application areas, it is often of interest to estimate a function nonparametrically subject only to certain constraints on its shape. Typical examples include (but are not limited to) monotone dose-response curves in medicine (Kelly and Rice 1990), concave utility functions in econometrics (Meyer and Pratt 1968), increasing growth curves, non-increasing survival function or 'U'-shaped hazard function (Reboul 2005) in survival analysis, computed tomography (Prince and Willsky 1990), target reconstruction (Lele et al. 1992), image analysis (Goldenshluger and Zeevi 2006), queuing theory (Chen and Yao 1993), and circuit design (Nicosia et al. 2008).

A Bayesian framework offers a unified probabilistic way of incorporating various shape constraints and accordingly there is a large literature devoted to Bayesian shape constrained estimation. A general approach is to expand the unknown function in a basis and translating the functional constraints to linear constraints in the coefficient space. Some representative examples include piecewise linear models (Neelon and Dunson 2004; Cai and Dunson 2007), Bernstein polynomials (Curtis and Ghosh 2011), regression splines (Meyer et al. 2011), penalized spines (Brezger and Steiner 2008), cumulative distribution functions (Bornkamp and Ickstadt 2009), and restricted splines (Shively et al. 2011) used as the basis. Gaussian process (GP) priors have also been employed for shape-restricted inference. Riihimäki and Vehtari (2010) proposed a method for imposing monotonicity information by including derivative observations in a GP model. More recently, Lin and Dunson (2014) proposed an approach based on projecting the posterior samples from an unrestricted GP fit to the constrained space.

In this article, we focus on a recent approach due to Maatouk and Bay (2017) who exploited a novel basis representation to equivalently represent various shape restrictions such as boundedness, monotonicity, convexity etc as non-negativity constraints on the basis coefficients. Although originally developed in the context of computer model emulation, the approach of Maatouk and Bay (2017) is broadly applicable to general shape constrained problems. Zhou et al. (2019) adapted their approach to handle a combination of shape constraints in a nuclear physics application to model the electric form factor of a proton. The main idea of Maatouk and Bay (2017) is to expand a Gaussian process using a first or second order exact Taylor expansion, with the remainder term approximated using linear combinations of compactly supported triangular basis functions. A key observation is that the resulting approximation has the unique advantage of enforcing linear inequality constraints on the

✉ Anirban Bhattacharya
anirbanb@stat.tamu.edu

[1] Department of Statistics, Texas A&M University, 3143 TAMU, College Station, TX 77843, USA

function space through an *equivalent* linear constraint on the basis coefficients. In terms of model fitting under a standard Gibbs sampling framework, this necessitates sampling from a high-dimensional truncated multivariate normal (tMVN) distribution.

The problem of sampling from a tMVN distribution is notoriously challenging in high dimensions and a number of solutions have been proposed in the literature. Existing Gibbs samplers for a tMVN distribution sample the coordinates one-at-a-time from their respective full conditional truncated univariate normal distributions (Geweke 1991; Kotecha and Djuric 1999; Damien and Walker 2001; Rodriguez-Yam et al. 2004). While the Gibbs sampling procedure is entirely automated, such one-at-a-time updates can lead to slow mixing, especially if the variables are highly correlated. More recently, Pakman and Paninski (2014) proposed a Hamiltonian Monte Carlo (HMC) algorithm which has drastically improved the speed and efficiency of sampling from tMVNs. However, implementing this algorithm within a larger Gibbs sampler can still lead to inefficiencies if the sample size is large. The second and third authors of this article encountered this challenge in Zhou et al. (2019) with a sample size greater than 1000. A related issue which contributes to the complexity is the $\mathcal{O}(N^3)$ computation and $\mathcal{O}(N^2)$ storage requirements for inverting and storing a general $N \times N$ covariance matrix.

In this article, we propose a novel algorithm to exploit additional structure present in the tMVN distributions arising in the aforesaid shape-constrained problems using the basis of Maatouk and Bay (2017). Our approach is based on a novel combination of elliptical slice sampling (ESS; Murray et al. 2010), circulant embedding techniques, and smooth relaxations of hard constraints. We additionally use Durbin's recursion to efficiently update hyperparameters within the covariance kernel of the parent Gaussian process. We analyze the per-iteration complexity of the proposed algorithm and illustrate through simulated and real data examples that the proposed algorithm provides significant computational advantages while retaining the statistical accuracy. `R` code to implement the proposed algorithm for monotone and convex function estimation is provided at https://github.com/raypallavi/BNP-Computations. We note that our algorithm and code be trivially adapted to other basis functions.

The rest of the paper is organized as follows. In Sect. 2, we revisit the Bayesian shape constrained function estimation problem and describe the novel algorithm for inference. The algorithm is specialized to estimating monotone and convex functions in Sect. 3. We provide a re-analysis of the proton dataset considered in Zhou et al. (2019) using our more efficient implementation in Sect. 4 and additional numerical illustrations on synthetic data are in Sect. 5. We conclude with a discussion in Sect. 6. Various algorithmic and implementation details are provided in an "Appendix".

2 Algorithm development

Consider the problem of sampling from a distribution having the following form:

$$p(\xi) \propto \exp\left\{-\frac{1}{2\sigma^2}\|Z-\Phi\xi\|^2\right\} \exp\left\{-\frac{1}{2\tau^2}\xi^{\mathrm{T}}K^{-1}\xi\right\} \mathbb{1}_{\mathcal{C}_\xi}(\xi), \quad \xi \in \mathbb{R}^N, \tag{1}$$

where $Z \in \mathbb{R}^n$, $\Phi \in \mathbb{R}^{n\times N}$ with $N \leq n$, $\mathcal{C}_\xi \subset \mathbb{R}^N$ is determined by a set of linear inequality constraints on ξ, and K is positive definite matrix. While our methodology generally applies to any such K, we are specifically interested in situations where K arises from the evaluation of a stationary covariance kernel on a regular grid.

The distribution (1) arises as a conditional posterior of basis coefficients in many Bayesian nonparametric regression problems where linear shape constraints (such as monotonicity, convexity, or a combination of these; Zhou et al. 2019) on the regression function are present, and a constrained Gaussian process prior is placed on the coefficient vector ξ in an appropriate basis representation. Sampling from the density (1) is then necessitated within a larger Gibbs sampler to fit the said constrained regression model.

Specifically, suppose we observe response-covariate pairs $(y_i, x_i) \in \mathbb{R} \otimes \mathbb{R}^d$ for $i = 1, \ldots, n$, related by the Gaussian regression model

$$y_i = f(x_i) + \epsilon_i, \quad \epsilon_i \sim \mathrm{N}(0, \sigma^2), \quad i = 1, \ldots, n \tag{2}$$

where the unknown regression function f is constrained to lie in some space $\mathcal{C}_f$, a subset of the space of all continuous functions on $[0, 1]^d$. When $\mathcal{C}_f$ corresponds to the space of monotone or convex functions, Maatouk and Bay (2017) identified a novel basis representation for f which allowed *equivalent representations* of the aforesaid constraints in terms of the basis coefficients ξ restricted to the positive orthant:

$$\mathcal{C}_\xi := \mathcal{C}_\xi^N = \left\{\xi \in \mathbb{R}^N : \xi_j \geq 0, \ j = 1, \ldots, N\right\}. \tag{3}$$

We provide more details on their basis representation in Sects. 3.1 and 3.2. They also considered the case where the regression function is globally bounded between two constants. See also Zhou et al. (2019) where a combination of interpolation, monotonicity, and convexity constraints can be equivalently expressed in terms of linear constraints on the coefficients.

Relating the basis coefficients ξ with the function values and (or) its derivatives, Maatouk and Bay (2017) proposed a constrained Gaussian prior on ξ. If the function f was uncon-

strained, then a Gaussian process (GP) prior on f induces a Gaussian prior on ξ, aided by the fact that derivatives of GP are again GPs provided the covariance kernel is sufficiently smooth. A natural idea then is to restrict the induced prior on ξ to the constrained region $\mathcal{C}_\xi$,

$$\pi(\xi) \propto \mathcal{N}(\xi; 0, \tau^2 K)\, \mathbb{1}_{\mathcal{C}_\xi}(\xi),$$

which is precisely the specification of Maatouk and Bay (2017). The density in Eq. (1) is then recognized as the conditional posterior of ξ.

In what follows, we shall additionally assume that $K_{jj'} = k(u_j - u_{j'})$ for a positive definite function k and a set of uniform grid points $\{u_j\}$. For example, if $d = 1$, we have $u_j = j/N$ for $j = 0, 1, \ldots, N$. This is a slight departure from Maatouk and Bay (2017) in the monotone and convex case. Since the derivatives of a stationary GP is generally non-stationary, so is their induced prior on ξ from a parent stationary GP on f. We instead directly place a stationary GP on an appropriate derivative of f, which results in K having a form as above. While there is little difference between the two approaches operationally, there is a large computational benefit for our approach, as we shall see below.

Returning to (1), a simple calculation yields that $p(\xi)$ is a truncated normal distribution, specifically,

$$\mathcal{N}_N\Big((\Phi^{\mathrm{T}}\Phi/\sigma^2 + K^{-1}/\tau^2)^{-1}\Phi^{\mathrm{T}}Y, (\Phi^{\mathrm{T}}\Phi/\sigma^2 + K^{-1}/\tau^2)^{-1}\Big),$$

truncated to $\mathcal{C}_\xi$. While one can use off-the-shelf samplers for tMVNs (Pakman and Paninski 2014) to sample from the above, the intrinsic complexity of sampling from tMVNs coupled with the computation and storage of the inverse of the kernel matrix K contributes to the challenges of executing this sampling step for large N. In particular, $(\Phi^{\mathrm{T}}\Phi/\sigma^2 + K^{-1}/\tau^2)$ keeps changing over each MCMC iteration with new updates of σ and τ, which requires an $N \times N$ matrix inversion at each iteration while applying any of the existing algorithms. The usual Sherman–Morrison–Woodbury matrix inversion trick does not render beneficial in this case. Barring issues with matrix inversions, implementation of such algorithm will be expensive in terms of storage. In addition, if there are unknown hyperparameters in the covariance kernel that get updated at each iteration within a larger MCMC algorithm, either the inversion has to take place at each step, or one has to pre-store a collection of K^{-1} on a fine grid for the hyperparameters.

In this article, we present a different approach to sample from the density p in (1) which entirely avoids matrix inversions. Our approach is based on three basic building blocks: (i) approximating the indicator function in p with a smooth approximant, (ii) a novel use of elliptical slice sampling (Murray et al. 2010) to avoid sampling from truncated non-Gaussian distribution, and (iii) using highly efficient samplers based on the fast Fourier transform for stationary GPs on a regular grid (Wood and Chan 1994). We describe the details below, starting with a brief review of elliptical slice sampling.

The elliptical slice sampler is a general technique for sampling from posterior distributions of the form,

$$p(\xi) \propto L(\xi)\,\mathcal{N}(\xi; 0, \Sigma)$$

proportional to the product of a zero-mean multivariate Gaussian prior with a general likelihood function $L(\cdot)$. In this context, Metropolis–Hastings proposals

$$\xi' = \rho\, \nu_e + \sqrt{1-\rho^2}\,\xi, \quad \nu_e \sim \mathcal{N}(0, \Sigma)$$

for $\rho \in [-1, 1]$ are known to possess good empirical (Neal 1999) and theoretical (Cotter et al. 2013) properties. Such an AR(1)-type proposal preserves the mean-zero $\mathcal{N}(0, \Sigma)$ prior, i.e., if $\xi \sim \mathcal{N}(0, \Sigma)$ and $\xi'|\xi \sim \mathcal{N}(\sqrt{1-\rho^2}\xi, \rho^2\Sigma)$ is drawn as above, then the marginal distribution of ξ' is again $\mathcal{N}(0, \Sigma)$. Using this fact, it is readily seen that the Metropolis–Hastings acceptance ratio $\alpha = \min\Big(1, L(\xi')/L(\xi)\Big)$ only depends on the likelihood ratio and is free of ρ. The elliptical slice sampler presents an adaptive and automated way to tune the step-size parameter ρ which guarantees acceptance at each step. Specifically, a new location on the randomly generated ellipse determined by the current state ξ and the auxiliary draw ν_e is produced according to

$$\xi' = \nu_e \sin\theta + \xi \cos\theta \tag{4}$$

where the angle θ is uniformly generated from a $[\theta_{\min}, \theta_{\max}]$ interval which is shrunk exponentially fast until an acceptable state is reached. To be precise, for each such θ, a uniform random number is drawn which compared against the likelihood ratio $L(\xi')/L(\xi)$. If the proposal ξ' is not acceptable, one shrinks the bracket of θ, and continues this process until acceptance; specific details of shrinking the bracket can be found in Murray et al. (2010). Hence, to extend the ESS algorithm by one step, the only requirement is to evaluate L at arbitrary points, which renders the approach broadly applicable.

Turning to (1), note however that the elliptical slice sampler is not immediately applicable as we have a truncated normal prior. As a simple fix-up, we approximate the indicator function $\mathbb{1}_{\mathcal{C}_\xi}(\cdot)$ in (1) by a suitable smooth function. Specifically, assuming $\mathcal{C}_\xi$ has the same structure as in (3), we use sigmoid-like approximations $\mathbb{1}_{(0,\infty)}(x) \approx (1 + e^{-\eta x})^{-1}$ for large $\eta > 0$ to obtain a smooth approximation $\mathbb{J}_\eta(\cdot)$ to $\mathbb{1}_{\mathcal{C}_\xi}(\cdot)$ as

$$\mathbb{1}_{\mathcal{C}_\xi}(\xi) \approx \mathbb{J}_\eta(\xi) = \prod_{j=1}^{N} \frac{e^{\eta\xi_j}}{1+e^{\eta\xi_j}}. \tag{5}$$

With $\mathbb{J}_\eta(\xi)$ defined like this, let us define

$$\begin{aligned}
\widetilde{p}(\xi \mid -) &\propto \exp\left\{-\frac{1}{2\sigma^2}\|Y-\Phi\xi\|^2\right\} \exp\left\{ \right. \\
&\left. -\frac{1}{2\tau^2}\xi^{\mathrm{T}}K^{-1}\xi\right\} \mathbb{J}_\eta(\xi) \\
&= \left[\exp\left\{-\frac{1}{2\sigma^2}\|Y-\Phi\xi\|^2\right\}\mathbb{J}_\eta(\xi)\right] \exp\left\{ \right. \\
&\left. -\frac{1}{2\tau^2}\xi^{\mathrm{T}}K^{-1}\xi\right\} \\
&= \left[\exp\left\{-\frac{1}{2\sigma^2}\|Y \right.\right. \\
&\left.\left. -\Phi\xi\|^2\right\}\left\{\prod_{j=1}^{N}\frac{e^{\eta\xi_j}}{1+e^{\eta\xi_j}}\right\}\right] \exp\left\{-\frac{1}{2\tau^2}\xi^{\mathrm{T}}K^{-1}\xi\right\}.
\end{aligned} \tag{6}$$

The density $\widetilde{p}$ defined on $\mathbb{R}^N$ approximates the density p. The parameter η controls the quality of the approximation; higher the value of η, better is the approximation. Experimenting across a large number of simulation scenarios, we find that $\eta = 50$ already provides an accurate approximation for dimensions at least up to 1000.

We now focus on sampling from $\widetilde{p}$ (using MCMC), which we shall consider as approximate samples from p. There is a growing literature on such approximate MCMC algorithms; see for example Bardenet et al. (2017); Johndrow et al. (2017) for more discussion and references. Later in Sect. 4, we device a strategy to modify the MCMC sampler for $\widetilde{p}$ into an MCMC with stationary distribution p.

We apply ESS to draw samples from $\widetilde{p}(\xi \mid -)$, since treating the quantity in the square brackets in (6) as "redefined likelihood", ξ has an (untruncated) multivariate Gaussian prior, which we call as the "working prior". Thus, one just needs to draw samples from the "working prior" distribution and compute the logarithm of the "redefined likelihood" function. In our case, computing the log-likelihood function has computational cost of $\mathcal{O}(nN)$ and we are to sample $\nu_e \sim \mathcal{N}(0, \tau^2 K)$, which is usually of $\mathcal{O}(N^3)$. Note that these computational complexities correspond to a single iteration of the MCMC sampler.

Under the assumption that the covariance matrix K is obtained from a regular grid in $[0, 1]$, sampling from the "working prior" is same as simulating realizations of a stationary Gaussian Process on a regular grid in $[0, 1]$. Such a covariance matrix is known to have a Toeplitz structure and the simulation can be carried out using the sampling scheme developed by Wood and Chan (1994) which reduces the overall complexity of the algorithm to a commendable extend. The details of this algorithm is discussed in the following section.

2.1 Sampling from the prior distribution of ξ

Sampling from the "working prior" distribution requires sampling from a stationary GP on a regular grid in $[0, 1]$ with a Toeplitz structure of the covariance matrix. In such settings, the algorithm of Wood and Chan (1994) based on clever embedding techniques can be readily applied. In particular, they exploit the discrete fast Fourier transform twice to offer substantially reduced compared cost. We briefly discuss some of the key ingredients of the algorithm.

The goal is to sample a random vector of the form

$$Z = \left(Z(0), Z\left(\frac{1}{m}\right), Z\left(\frac{2}{m}\right), \ldots, Z\left(\frac{m-1}{m}\right)\right)^{\mathrm{T}}$$

from a mean-zero Gaussian random process on each of the grid points

$$\begin{aligned}
\left\{0, \frac{1}{m}, \frac{2}{m}, \ldots, \frac{m-1}{m}\right\} &\equiv \left\{u_j : u_j \right. \\
&\left. = \frac{j}{m}; 0 \le j < 1\right\}, \quad m \ge 1
\end{aligned}$$

with covariance function $\gamma : \mathbb{R} \to \mathbb{R}$. Then $Z \sim \mathcal{N}_m(0, G)$, where

$$G = \begin{bmatrix}
\gamma(0) & \gamma\left(\frac{1}{m}\right) & \cdots & \gamma\left(\frac{m-1}{m}\right) \\
\gamma\left(\frac{1}{m}\right) & \gamma(0) & \cdots & \gamma\left(\frac{m-1}{m}\right) \\
\vdots & \vdots & \ddots & \vdots \\
\gamma\left(\frac{m-1}{m}\right) & \gamma\left(\frac{m-2}{m}\right) & \cdots & \gamma(0)
\end{bmatrix}$$

It is to be noted that G is a Toeplitz matrix, which is equivalent to $\tau^2 K$ in our notation.

There are two basic steps of this method:

1. Embedding G in a circulant covariance matrix C of order $d \times d$, where $d = 2^g$, for some integer g and $d \ge 2(m-1)$. The circulant matrix is formed such a way that by construction, C is symmetric and the $m \times m$ submatrix in the top left corner of C is equal to G. For details on the embedding technique, one can refer to Wood and Chan (1994).
2. Using fast Fourier transform **twice** to generate $Z = \left(Z_0, Z_1, \ldots, Z_{d-1}\right) \sim \mathcal{N}_d(0, C)$. Then due to appropriate construction of C, $\left(Z_0, Z_1, \ldots, Z_{m-1}\right) \sim \mathcal{N}_m(0, G)$. Note that C needs to be positive definite.

In summary, Wood and Chan (1994) essentially uses a parameter-expansion scheme to translate the problem of sam-

pling from $\mathcal{N}_m(0, G)$ to sampling $\mathcal{N}_d(0, C)$ for some $d > m$. Exploiting the fact that the covariance matrix C of the larger Gaussian is a symmetric circulant matrix, the task of sampling from $\mathcal{N}_d(0, C)$ can be accomplished $\mathcal{O}(d \log d)$ steps. This sampling scheme exploits efficient computation of the eigenvalues of C based on the powerful FFT algorithm; exact details can be found in §5 of Wood and Chan (1994). Thus, even though $d > m$, the above implementation can provide large speed-ups provided $d \ll m^3$. This is the case we observe throughout our numerical studies.

It is to be noted that the circulant matrix C is not guaranteed to be positive definite for any $d \geq 2(m-1)$. We followed the exact approach of Wood and Chan (1994) to search for the smallest d which makes C positive definite. In situations where such a d cannot be found or is too large to be practicable, Wood and Chan (1994) suggested an approximate scheme to make C nonnegative definite. We did not encounter the need to pursue this approximation scheme for the scale of problems we considered here, although this may come handy for larger datasets.

2.2 Algorithm

We implemented our algorithm with K as a stationary Mátern kernel with smoothness parameter $\nu > 0$ and length-scale parameter $\ell > 0$. Our method takes design points X, observations Y, Mátern kernel parameters ν and ℓ, η as in (6), dimension of the random coefficients N and number of posterior samples n_0 as inputs and gives n_0 many posterior samples of ξ as output.

Algorithm 1 Efficient algorithm to draw posterior samples of ξ

Input: $X, Y, \nu, \ell, \eta, N, \tau^2, \sigma^2$ and n_0
Using N, calculate $u_j = j/N$, $j = 0, \ldots, N$;
Using X and u_j's form basis matrix Φ
Using ν, ℓ and u_j's form covariance matrix K
Initialize : $\xi^{(0)}$
for $t = 1$ **to** n_0 **do**
  Sample $v_e \sim \mathcal{N}(0, \tau^2 K)$ using simulation scheme by Wood and Chan (1994).
  Sample $\xi^{(t)}$ using v_e, η and σ^2 following ESS scheme by Murray et al. (2010).
end for
Output: Posterior samples of ξ of size n_0.

The computation cost for drawing a random sample from the prior distribution usually dominates. But that is not the case here. Since $n > N$, computational cost for computing the log-likelihood, using ESS scheme, dominates which leads to computational complexity of $\mathcal{O}(nN)$, for each MCMC iteration.

2.3 Updating hyperparameters

As already discussed, updating the hyperparameters present in the covariance matrix K is computationally challenging in the absence of any structure in K. Any likelihood-based method for updating ν and ℓ (e.g. Metropolis–Hastings) requires computing K^{-1} which leads to $\mathcal{O}(N^3)$ computational steps and $\mathcal{O}(N^2)$ storage. Hence the computational complexity per MCMC iteration of Algorithm 1 is always bounded above by $\mathcal{O}(N^3)$.

However, substantial speed-up is possible in our case as K is a symmetric positive-definite Toeplitz matrix. We turn to a non-trivial but effective approach of finding K^{-1} utilizing inverse Cholesky factor of K using Durbin's recursion algorithm (Golub and van Loan 1996) which has a computational complexity of $\mathcal{O}(N^2)$. The columns of the inverse Cholesky factor of K is obtained by solving Yule–Walker systems. Durbin recursion is a procedure of recursively finding the solution to a system of equations involving a Toeplitz matrix, in particular, it is applicable to Yule–Walker systems. Given real numbers $r_0, r_1, \ldots, r_{M-1}$ with $r_0 = 1$ such that $T = \left(r_{|i-j|}\right) \in \mathbb{R}^{M \times M}$ is positive definite then Durbin's algorithm computes $u \in \mathbb{R}^M$ as a solution of the Yule–Walker problem:

$$T\, u = -(r_1, \ldots, r_{M-1})^{\mathrm{T}}$$

For more details on Durbin's recursion for solving Yule–Walker equation, refer to Golub and van Loan (1996).

Now suppose, we have the Cholesky factor R such that $R^{\mathrm{T}} R = T$ where R is an upper-triangular matrix and the inverse Cholesky factor is given by R^{-1}. Therefore, $T R^{-1} = R^{\mathrm{T}}$ and noting that R^{T} is lower-triangular, it is enough to solve only the upper-triangular part of R^{-1}. The first h elements of the hth column of R^{-1} can be found as a solution of

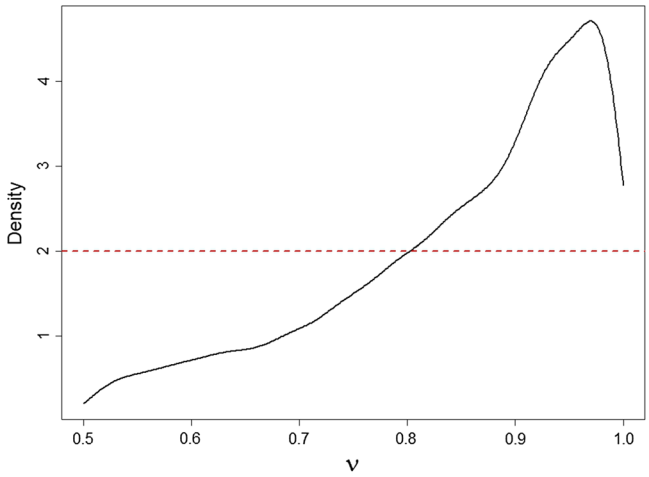
$$T_h\, u^{(h)} = -r^{(h)}, \quad h = 1, \ldots, M$$

where T_h is the $h \times h$ principal submatrix of T, $u^{(h)}$ is h-dimensional vector of solutions and $r^{(h)} = (r_1, \ldots, r_h)^{\mathrm{T}}$ and each of these M equations can be solved using Durbin's algorithm mentioned above. Note that, the hth column of R^{-1} denoted by $(R^{-1})_h$ is then given by:

$$(R^{-1})_h = \begin{bmatrix} E_h & O_{h\times(M-h)} \\ O_{(M-h)\times h} & O_{(M-h)\times(M-h)} \end{bmatrix} \begin{bmatrix} u^{(h)} \\ O_{(M-h)\times h} \end{bmatrix}$$

where E_h is an *exchange matrix* of order h with anti-diagonal elements all ones and other elements are all zeros. This approach requires $\mathcal{O}(M^2)$ computations to find R^{-1}.

We considered continuous uniform priors on compactly supported intervals on both ν and ℓ, independently of each



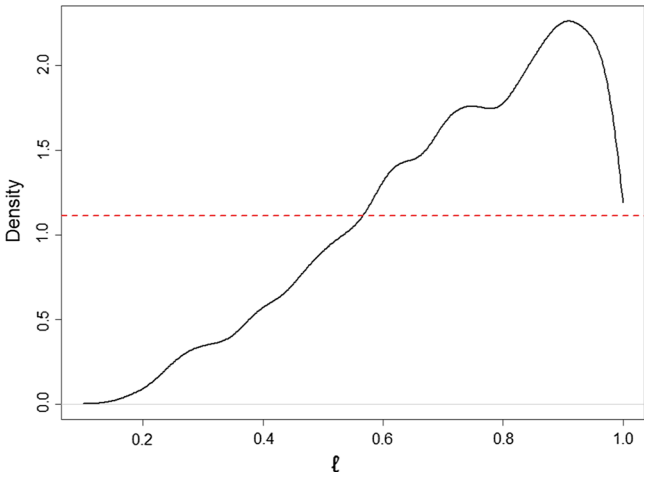


Fig. 1 Posterior density plots of the hyperparameters ν (left panel) and ℓ (right panel) represented by solid black curves, and the prior densities given by dotted red lines. Support of ν is [0.5, 1] and that for ℓ is [0.1, 1]. Posterior samples were drawn using Metropolis–Hastings and utilizing S obtained through Durbin's recursion

other. Updating $\nu \mid \xi, -$ and $\ell \mid \xi, -$ using Metropolis–Hastings requires to compute acceptance ratio which involves computation of $\xi^{\mathrm{T}} K^{-1} \xi$ and $|K|^{-1/2}$ for proposal and current combinations of (ν, ℓ). Using Durbin's algorithm, we can find S such that $(S^{-1})^{\mathrm{T}} S^{-1} = K$ and then $\xi^{\mathrm{T}} K^{-1} \xi = (S^{\mathrm{T}} \xi)^{\mathrm{T}} (S^{\mathrm{T}} \xi) = \sum_{j=1}^{N} v_j$ where $v = S^{\mathrm{T}} \xi$ and $|K|^{-1/2} = \prod_{j=1}^{N} S_{jj}$. Evidently, computation for S dominates and Durbin's algorithm allows us to update the hyperparameters in $\mathcal{O}(N^2)$ computations for each iteration within an MCMC algorithm. Thus per-iteration computational complexity for Algorithm 1 combined with this hyperparameter update technique remains $\mathcal{O}(nN)$ as before.

Figure 1 shows the posterior density plots of ν (left panel) and ℓ (right panel) and comparison with the uniform prior densities. We generated 500 paired data of response and covariate based on (2) with $\sigma = 0.05$ and the true data generating function f is monotone, given by $f(x) = \log(20x+1)$. Posterior samples were drawn using Metropolis–Hastings and the previously mentioned computation scheme. Posterior densities suggest that it is possible to learn the hyperparameters through this technique. Moreover, based on numerous simulation studies that we had conducted, the Metropolis–Hastings sampler for the hyperparameter update attained at least 15% acceptance probability.

3 Application to shape constrained estimation

We now return to the constrained Gaussian regression setup in (2), and consider applications of our sampling algorithm to situations when f is a smooth monotone or convex function.

We first introduce some notation to define the basis functions employed by Maatouk and Bay (2017).

Let $\{u_j \in [0,1],\ j = 0, 1, \ldots, N\}$ denote equally spaced knots on [0, 1] with spacing $\delta_N = 1/N$ and $u_j = j/N$. Let

$$h_j(x) = h\left(\frac{x - u_j}{\delta_N}\right), \quad \psi_j(x) = \int_0^x h_j(t)\, dt, \quad \phi_j(x) = \int_0^x \int_0^t h_j(u)\, du\, dt; \quad x \in [0,1]$$

where $h(x) = (1 - |x|)\, \mathbb{1}_{[-1,1]}(x)$. The collection of functions $\{h_j\}$ is called the *interpolation basis* by Maatouk and Bay (2017), since for any continuous function $f : [0,1] \to \mathbb{R}$, the function $\widetilde{f}(\cdot) = \sum_{j=0}^{N} f(u_j)\, h_j(\cdot)$ approximates f by linearly interpolating between the function values at the knots $\{u_j\}$.

The integrated basis $\{\psi_j\}$ and $\{\phi_j\}$ take advantage of higher-order smoothness. For example, if f is continuously differentiable, then by the fundamental theorem of calculus,

$$f(x) - f(0) = \int_0^x f'(t) dt.$$

Expanding f' in the interpolation basis implies the model

$$f(x) = \xi_0 + \sum_{j=0}^{N} \xi_{j+1} \psi_j(x). \tag{7}$$

Similarly, if f is twice continuously differentiable, we have

$$f(x) - f(0) - x f'(0) = \int_0^x \int_0^t f''(s)\, ds dt.$$

Now expanding f' and f'' in the interpolation basis implies the model

$$f(x) = \xi_0 + \xi^* x + \sum_{j=0}^{N} \xi_{j+1} \phi_j(x). \tag{8}$$

Maatouk and Bay (2017) showed that under (7), f is monotone non-decreasing *if and only if* $\xi_i \geq 0$ for all $i = 1, \ldots, N+1$. Similarly, under (8), f is convex non-decreasing *if and only if* $\xi_i \geq 0$ for all $i = 1, \ldots, N+1$. This equivalence relationship between the functional constraint and the linear inequality constraints on the basis coefficients is an attractive feature of the interpolation basis and is not shared by many commonly used basis functions.

For an unrestricted f, a GP prior on f implies a dependent Gaussian prior on the coefficient vector $\xi = (\xi_1, \ldots, \xi_{N+1})^{\mathrm{T}}$. A natural idea is to restrict this dependent prior subject to the linear restrictions on the coefficients which results in a dependent tMVN prior. Fitting the resulting model using a Gibbs sampler, the full conditional of ξ assumes the form (1), rendering our Algorithm 1 applicable.

We provide more details regarding the model and prior for the monotone and convex cases separately. Let $X = (x_1, \ldots, x_n)^{\mathrm{T}}$ be the vector of n design points, $Y = (y_1, \ldots, y_n)^{\mathrm{T}}$ be the vector of corresponding responses.

3.1 Monotonicity constraint

We can express (7) in vector notation as

$$Y = \xi_0 1_n + \Psi \xi + \varepsilon, \quad \varepsilon \sim \mathcal{N}_n(0, \sigma^2 \mathrm{I}_n), \quad \xi \in \mathcal{C}_\xi^{N+1}, \tag{9}$$

where recall from (3) that $\mathcal{C}_\xi^m$ denotes the positive orthant in $\mathbb{R}^m$ and $\xi = (\xi_1, \ldots, \xi_{N+1})^{\mathrm{T}}$. Also, Ψ is an $n \times (N+1)$ basis matrix with ith row Ψ_i^{T} where $\Psi_i = (\psi_0(x_i), \ldots, \psi_N(x_i))^{\mathrm{T}}$ and 1_n denotes an n dimensional vector of all 1's.

The parameter $\xi_0 \in \mathbb{R}$ is unrestricted, and we place a flat prior $\pi(\xi_0) \propto 1$ on ξ_0. We place a tMVN prior on ξ independently of ξ_0 as $p(\xi) \propto \mathcal{N}(\xi; 0, \tau^2 K)\, \mathbb{1}_{\mathcal{C}_\xi}(\xi)$, where $K = (K_{jj'})$ with $K_{jj'} = k(u_j - u_{j'})$ and $k(\cdot)$ the stationary Mátern kernel with smoothness parameter $\nu > 0$ and length-scale parameter $\ell > 0$. To complete the prior specification, we place improper priors $\pi(\sigma^2) \propto 1/\sigma^2$; $\pi(\tau^2) \propto 1/\tau^2$ on σ^2 and τ^2, and compactly supported priors $\nu \sim \mathcal{U}(0.5, 1)$ and $\ell \sim \mathcal{U}(0.1, 1)$ on ν and ℓ. A straightforward Gibbs sampler is used to sample from the joint posterior of $(\xi_0, \xi, \sigma^2, \tau^2, \nu, \ell)$ whose details are deferred to the Appendix. The parameters σ^2, τ^2 and ξ_0 have standard conditionally conjugate updates. The key feature of our algorithm is sampling the high-dimensional parameter ξ using Algorithm 1 and updating ν and ℓ via Metropolis-within-Gibbs using Durbin's recursion as outlined in Sect. 2.3.

Before concluding this section, we comment on a subtle difference in our prior specification from Maatouk and Bay (2017), which nevertheless has important computational implications. Since the basis coefficients ξ_j, $j \geq 1$ target the derivatives $f'(u_j)$, Maatouk and Bay (2017) consider a joint prior on (ξ_0, ξ) obtained by computing the induced prior on $\big(f(0), f'(u_0), \ldots, f'(u_N)\big)$ from a GP prior on f, and then imposing the non-negativity restrictions. Since the derivative of a sufficiently smooth GP is again a GP, the joint distribution of $\big(f(0), f'(u_0), \ldots, f'(u_N)\big)$ can be analytically calculated. However, one downside is that the derivative of a stationary GP is no longer stationary in general, and thus sampling from the joint Gaussian prior of $\big(f(0), f'(u_0), \ldots, f'(u_N)\big)$ cannot take advantage of the embedding techniques for a stationary GP. We instead directly place a prior on ξ induced from a stationary GP prior on f' and then imposing the necessary restrictions. Since ξ_0 is only a single real-valued parameter, we break the dependence between ξ_0 and ξ in the prior and assign a flat prior on ξ_0 independent of ξ. Although not reported here, our simulations suggest against any loss of efficiency in doing so, while there is a substantial computational gain because Algorithm 1 becomes readily applicable to update ξ with our prior. Alternatively, one may also consider a joint prior on (ξ_0, ξ) by working out the joint covariance structure of $\big(f(0), f'(u_0), \ldots, f'(u_N)\big)$, where f here stands for the anti-derivative of f', which again follows a Gaussian process. An illustration of such a joint prior is provided in Sect. 4.

3.2 Convexity constraint

The development here proceeds in a similar fashion to the monotone case and we only provide a brief sketch. We can write (8) in vector notation as

$$Y = \xi_0 1_n + \xi^* X + \Phi \xi + \varepsilon, \quad \varepsilon \sim \mathcal{N}_n(0, \sigma^2 \mathrm{I}_n), \quad \xi \in \mathcal{C}_\xi^{N+1}, \tag{10}$$

where Φ is an $n \times (N+1)$ basis matrix with ith row Φ_i^{T} and $\Phi_i = (\phi_0(x_i), \ldots, \phi_N(x_i))^{\mathrm{T}}$. The only additional parameter here from the previous case is ξ^* to which we assign a flat prior on ξ^* independent of everything else. For all other parameters, the exact same prior specification is employed as in the previous case. The details of the Gibbs sampler are once again deferred to the Appendix.

3.3 Run-time comparison

We empirically illustrate the computational complexity of our algorithm relative to a state-of-the-art method. We consider two examples corresponding to a monotone and convex truth respectively. For the monotone case, the true function $f(x) = \log(20x + 1)$, also considered in Maatouk and Bay

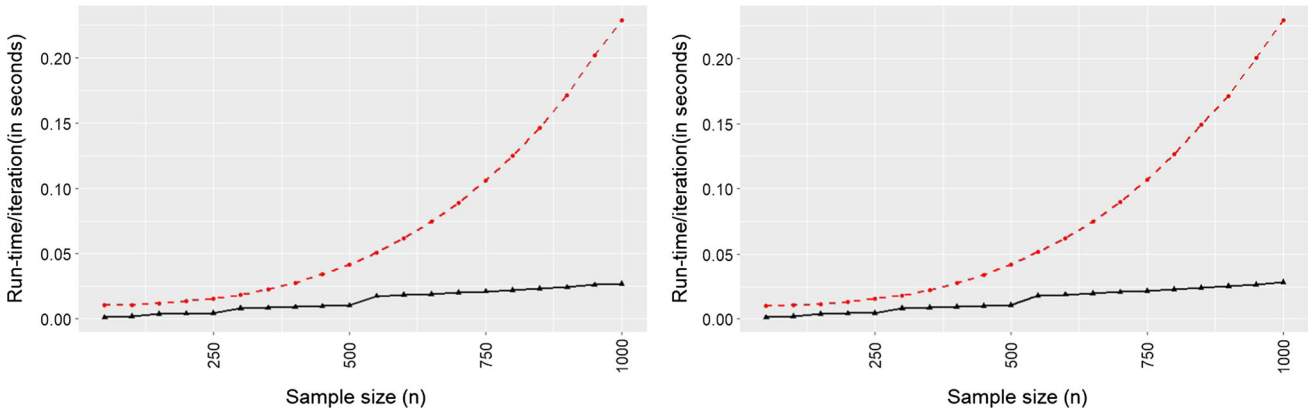


Fig. 2 Run-time per iteration (in seconds) against the sample size for two Gibbs samplers which only differ in the update of ξ in the monotone (left panel) and convex (right panel) estimation context. Our Algorithm 1 is represented by black triangles with solid black line while the `tmg` sampler is in red dots with dashed red line. (Color figure online)

(2017), while in the convex case, the true $f(x) = 5(x-0.5)^2$. In both cases, we uniformly generated the covariates on $[0, 1]$ and added Gaussian noise. We fixed $\eta = 50$, the number of knots to be half the sample-size, $N = \lceil n/2 \rceil$, $\nu = 0.75$ and ℓ was chosen so that the correlation at a maximum possible separation between the covariates equals 0.05. With $N = \lceil n/2 \rceil$, the computational complexity of Algorithm 1 within a single iteration of MCMC sampler is $\mathcal{O}(n^2)$.

We consider an alternative Gibbs sampler which samples $\xi \in \mathcal{C}_\xi^{N+1}$ from its tMVN full-conditional using the Hamiltonian Monte Carlo (HMC) sampler of Pakman and Paninski (2014), implemented in the **R** package "`tmg`". Keeping the hyperparameters fixed, all other parameters are updated in the exact same way in either sampler. We did not consider the rejection sampler used by Maatouk and Bay (2017) as it becomes quite inefficient with increasing dimension, with the "`tmg`" sampler substantially more efficient than the rejection sampler in high dimensions. The combination of Maatouk and Bay (2017) with the "`tmg`" sampler does not exist in the literature to the best of our knowledge, and thus we are being entirely fair in constructing the best possible competing method.

Figure 2 plots the run-time per iteration (in seconds) against the sample size n (varied between 50 and 1000) for the two approaches, both implemented under identical conditions on a quadcore Intel Core i7-2600 computer with 16 GB RAM. Evidently, Algorithm 1 provides more pronounced improvements for larger N.

4 Analysis of the proton puzzle problem using an exact version of our algorithm

The "proton radius puzzle" (Pohl et al. 2010; Bernauer and Pohl 2014; Carlson 2015) in Nuclear Physics refers to major inconsistencies regarding the extraction of the charge radius of a proton from different experimental procedures. The puzzle originated in 2010 when a newer suite of high-precision muonic Lamb-shift experiments suggested a value of the radius (0.8408 fm; 1 fm = 10^{-15} m) which is significantly (by $\sim$4%) different from the accepted 2010 Committee on Data for Science and Technology (CODATA) value (0.8775 fm) for the charge radius of the proton, arrived at using electron scattering and atomic spectroscopy experiments. The newer muonic measurements are known to be remarkably precise, and hence the fact that it hinted at such a major discrepancy caused shocking surprise.

Among a variety of approaches undertaken to explain the puzzle, a prominent school-of-thought (Higinbotham et al. 2016; Yan et al. 2018) is that the puzzle lies in the extraction of the radius from the old electron scattering experiments. The proton charge radius is related to the slope of the *form factor curve* at the origin, the electric form factor $G_E(Q^2)$ viewed as a function of the momentum transfer Q^2. Due to experimental limitations, the form factor curve cannot be observed at Q^2 values arbitrarily close to zero, and hence a subtle extrapolation to $Q^2 = 0$ is unavoidable. Therefore, the extraction of the charge radius from scattering data becomes a problem of estimating a curve (and its derivative) from noisy data.

The form factor curve is constrained by physical theory to be convex decreasing, with a known-value at the origin, $G_E(Q^2 = 0) = 1$. Recently, Zhou et al. (2019) developed a nonparametric Bayesian approach to model the form factor curve which obeys the above constraints and is otherwise flexible, unlike various models like monopole, dipole, etc previously used in this literature which make strong parametric assumptions. The estimated radius by Zhou et al. (2019) strongly supported a value of the radius close to 0.84 consistent with the muonic experiments; in fact, the old value of

0.87 was not contained in their 95% credible interval. Their analysis additionally demonstrated a substantial impact of incorporating the physical constraints on the form factor—without the constraints, a much wider interval is obtained which fails to differentiate between 0.87 and 0.84 fm.

Zhou et al. (2019) considered a Gaussian model (2), with the input variable $x = Q^2/Q^2_{\text{max}}$ being the momentum transfer Q^2 scaled to lie in [0,1], where Q^2_{max} denotes the maximum available Q^2 value. The mean function f, related to the form factor curve G_E through the equation $G_E(Q^2) = f(Q^2/Q^2_{\text{max}})$, is then modeled as in (8). In this parameterization, the quantity of interest, i.e., the proton radius $r_p := \sqrt{-6\xi^*}/Q_{\text{max}}$. Under the representation (8), they proved that the multiple restrictions on the form factor curve can be equivalently represented as linear equality and inequality constraints on the basis coefficients as

$$\left\{\xi_0 = 1,\ \xi^* + \sum_{j=0}^{N} c_j \xi_{j+1} \le 0,\ \xi_{j+1} \ge 0,\ j = 0, \ldots, N\right\}. \tag{11}$$

In the above display, $c_j = \psi_j(1)$ for all j.

Zhou et al. (2019) fixed $\xi_0 = 1$ and considered a tMVN prior $\mathcal{N}(0, \tau^2\Gamma)$ on (ξ^*, ξ) restricted to the region given by the inequality constraints in (11). Here, Γ is the covariance matrix of $\big(f'(0), f''(u_0), \ldots, f''(u_N)\big)$ induced from a stationary GP on f with a Mátern kernel with the smoothness parameter $\nu = 2.5$ to ensure twice-differentiable sample paths. As discussed in the third paragraph of Sect. 3.1, the derivatives of a GP with a stationary Mátern kernel are no longer stationary, and hence the matrix Γ does not inherit a Toeplitz structure. We instead follow our general prescription in Sect. 3.1 to begin with a stationary GP (with a Mátern kernel) on f'', with the smoothness parameter ν set to 0.5. The induced joint distribution[1] of $\big(f'(0), f''(u_0), \ldots, f''(u_N)\big)$ is $\mathcal{N}_{N+2}(0, \tau^2\Omega)$, where the lower $(N+1)\times(N+1)$ block of Ω has a Toeplitz structure. We then finally set the joint prior on (ξ^*, ξ) to be $\mathcal{N}(0, \tau^2\Omega)$ restricted to the region given by the inequality constraints in (11). Note that unlike Sect. 3.1, we cannot specify independent priors on ξ^* and ξ here since the dependence between them is already forced through the form of the constraints. We complete the prior specification with our default prior choices on σ^2 and τ^2 as before.

Under our prior specification, the conditional posterior of ξ is proportional to

$$\exp\left\{-\frac{1}{2\sigma^2}\|(Y - 1_n) - \xi^* X - \Phi\xi\|^2\right\} p(\xi^* \mid \xi)\, \mathcal{N}_{N+1}(\xi; 0, \tau^2\Omega_{22})\, \mathbb{1}_{\mathcal{C}^{N+1}_{\xi}}(\xi),$$

where Ω_{22} is the lower $(N+1)\times(N+1)$ block of Ω and $p(\xi^* \mid \xi)$ is the conditional density of $\xi^* \mid \xi$ from a joint $\mathcal{N}(0, \tau^2\Omega)$ density on (ξ^*, ξ). Recall that Ω_{22} has a Toeplitz structure due to our prior choice, so that sampling from $\mathcal{N}(0, \tau^2\Omega_{22})$ is efficient. We can thus make the sigmoid approximation to the indicators as before and proceed with the elliptical slice sampling to sample ξ. The quantity $p(\xi^* \mid \xi)$ is absorbed into the working likelihood; it's evaluation is cheap since it requires evaluation of a univariate normal density with mean $\Omega_{12}\Omega_{22}^{-1}\xi$ and variance $\tau^2(\Omega_{11} - \Omega_{12}\Omega_{22}^{-1}\Omega_{21})$. The only expensive matrix operation of $\Omega_{12}\Omega_{22}^{-1}$ can be performed once outside the MCMC loop and reused.

Due to the very fine nature of the inference problem involved, we correct for our approximation using a Metropolis–Hastings correction based on the following observation.

Proposition 4.1 *Let γ and γ_ε be probability densities on $\mathbb{R}^d$ with support$(\gamma) \subseteq$ support(γ_ε). Let $q_\varepsilon(\cdot,\cdot)$ be a Markov transition kernel which is reversible with respect to γ_ε (this in particular implies that γ_ε is the stationary or invariant distribution of q_ε). Define a new Markov transition kernel q which given a current state x, generates a proposal $x' \sim q_\epsilon(x, \cdot)$ and accepts it with probability*

$$\alpha(x, x') = \min\left\{1, \frac{w(x')}{w(x)}\right\}, \quad w(t) = \frac{\gamma(t)}{\gamma_\varepsilon(t)} \text{ for } t \in \text{support}(\gamma_\varepsilon).$$

If the move is not accepted, the chain stays at x. Then, the stationary distribution of $q(\cdot,\cdot)$ is γ.

Proof The transition kernel q can be expressed as

$$q(x, x') = \alpha(x, x')\, q_\varepsilon(x, x') + r(x)\, \delta_x(x'), \quad r(x) = \int \big(1 - \alpha(x, x')\big)\, q_\varepsilon(x, x') dx'.$$

We exhibit that q is reversible with respect to γ, i.e., the detailed balance condition $\gamma(x)\, q(x, x') = \gamma(x')\, q(x', x)$ holds for all $x, x' \in$ support(γ). This follows by noting that $\gamma(x)\, r(x)\delta_x(x') = \gamma(x')r(x')\delta_{x'}(x)$ is trivially true, and

$$\gamma(x)\, \alpha(x, x')\, q_\varepsilon(x, x') = \big(w(x)\, \alpha(x, x')\big)\, \big(\gamma_\varepsilon(x)\, q_\varepsilon(x, x')\big) = \min\{w(x'), w(x)\}\, \big(\gamma_\varepsilon(x')\, q_\varepsilon(x', x)\big) = \gamma(x')\, \alpha(x', x)\, q_\varepsilon(x', x).$$

Here we used the definition of $\alpha(\cdot,\cdot)$ and $w(\cdot)$ along with the reversibility of q_ε with respect to γ_ε. The stationarity then follows as an immediate consequence of reversibility. □

Proposition 4.1 provides a simple algorithm to turn an MCMC scheme to sample from γ_ε into a sampler for γ. Although the algorithm is valid for any choice of γ_ε and γ satisfying the assumptions, we are interested in situations where

[1] Here, f' should be interpreted as the anti-derivative of f'', i.e., $f'(x) = \int_0^x f''(t)dt$.

Table 1 Analysis of the proton data

N	$\widehat{r}_p$ (in fm)	CI_L (in fm)	CI_U (in fm)	Effective sample size	MCMC standard error	Average acceptance rate	Run-time/iter (in s) our algorithm	Ratio of run-time/iter for "tmg" versus our algo.
178	0.846	0.838	0.850	479.17	0.0001	0.997	0.02	2.50
356	0.846	0.839	0.849	479.30	0.0001	0.992	0.04	3.75
711	0.845	0.838	0.849	480.01	0.0001	0.997	0.08	13.13

Results corresponding to different choices of N using the exact version of our algorithm. $\widehat{r}_p$ is the estimate of posterior mean, CI_L and CI_U give the lower and upper limits of 95% CI, "Run-time/iter" represents run-time per MCMC iteration and are reported in seconds

γ_ε approximates γ as $\varepsilon \to 0$. Some immediate observations from Proposition 4.1 are in order. First, if the algorithm is initialized inside support(γ), then it always stays inside support(γ) since $w(t) = 0$ for $t \in \text{support}(\gamma_\epsilon)\backslash\text{support}(\gamma)$. Second, the implementation of the algorithm only requires knowing the densities γ and γ_ε upto normalizing constants. Finally, the smaller the value of ε (i.e., the better the approximation), the higher the acceptance rate will be on average.

For our purpose, we set $\gamma(\xi) \propto L(\xi)\, \mathbb{1}_{\mathcal{C}_\xi}(\xi)\, e^{-\xi^\text{T}\Omega_{22}^{-1}\xi/2}$ and $\gamma_\varepsilon(\xi) \propto L(\xi)\, \mathbb{J}_\eta(\xi)\, e^{-\xi^\text{T}\Omega_{22}^{-1}\xi/2}$, where $\mathbb{J}_\eta(\xi)$ is the sigmoid approximation defined in (5) and $\varepsilon = \eta^{-1}$. Reversibility of the elliptical slice sampler was proved in Murray et al. (2010), rendering Proposition 4.1 applicable. At each step of the MCMC, we proceed exactly as before to generate a draw ξ' using the ESS algorithm on γ_ε. While in our approximate version of the algorithm we always moved to ξ' from the current state ξ, we make the move with probability $\alpha(\xi, \xi')$ in the exact version. The acceptance ratio $\alpha(\xi, \xi')$ takes the simple form

$$\alpha(\xi, \xi') = \min\left\{1, \frac{\mathbb{1}_{\mathcal{C}_\xi}(\xi')/\mathbb{J}_\eta(\xi')}{\mathbb{1}_{\mathcal{C}_\xi}(\xi)/\mathbb{J}_\eta(\xi)}\right\}.$$

The sampling steps for the remaining univariate parameters are straightforward and very similar to those provided in the Appendix for the convex function case; hence we omit these steps.

We now report our analysis for the proton puzzle problem using the above exact version of our algorithm and the same Mainz dataset (Bernauer and Collaboration 2011; Bernauer et al. 2011, 2014) that Zhou et al. (2019) analyzed. We used 3 values for the number of knots $N \in \{\lceil n/8\rceil, \lceil n/4\rceil, n/2\}$, where $n = 1422$ is the number of observations. As noted earlier, we fixed $\nu = 0.5$ to maintain compatibility with Zhou et al. (2019) who used $\nu = 2.5$; recall that their prior operates on the original function while ours is at the level of the second derivative, thus the difference of 2 between our choice of ν from theirs. We fixed $\ell = 0.33$ so that the correlation between the two farthest knot points equals 0.05. We also tried $\ell = 0.5$ as suggested by Zhou et al. (2019) based on predictive cross-validation under their method. We omit the details for this case as the results were very close to our choice of $\ell = 0.33$.

We ran our sampler for 110,000 MCMC iterations, the first 10,000 of which were discarded as burn-in, and every 10th subsequent observation was stored. The second column of Table 1 shows the point estimates for the proton radius r_p for the different choices of N, while the third and fourth column respectively correspond to the upper and lower 95% symmetric credible intervals. Overall, our estimates were in close agreement with those obtained by Zhou et al. (2019) in their Table 1, and suggest a value of the radius around 0.84 fm. Moreover, the old value of 0.87 fm wasn't contained in a 95% credible interval under any scenario. The estimates also showed little sensitivity to the different choices for the number of knots employed.

We report the effective sample sizes and the MCMC standard error for r_p in the next two columns, which overall seem reasonable. These were calculated using the "mcmcse" package using the overlapping batch mean option and the theoretically optimal cubic root batch size. The next column shows the average acceptance rate (across the MCMC path) of the Metropolis correction in Proposition 4.1 to be very close to one across all scenarios, reaffirming that $\eta = 50$ provides a very accurate approximation in (6). Also, the acceptance probability for $\eta = 50$ is robust across various choices of the number of knots used.

The penultimate column reports the per-iteration run-time (in seconds) of our algorithm and the last column reports the ratio between the per-iteration run-times of the "tmg" sampler and ours. Zhou et al. (2019) used the **R** package "TruncatedNormal" to draw samples from the tMVN conditional posterior of ξ, which implements the exact rejection sampler due to Botev (2017). We replaced this step in their algorithm with the faster Hamiltonian Monte Carlo (HMC) sampler of Pakman and Paninski (2014), implemented in the **R** package "tmg". This step was incorporated to constructing the best possible competing method. We see a substantial time gain over "tmg", in particular, more than 10-times per-iteration speed-up for $N = 711$. While using the "tmg" sampler produces slightly better effective sam-

Table 2 Results corresponding to $n = 500$ and $N = \lceil n/8 \rceil = 63$ for different choices of ℓ using the exact version of our algorithm (abbreviated as "*Cons*") and the unconstrained counterpart of our method (abbreviated as "*unCons*")

$N = 63$		Coverage probability		Estimated radius		$\widehat{\mathbb{E}}(\lvert\widehat{r}_p - r_p\rvert)$	Length of 95% CI		Length 95% HPDI	
		95% CI	95% HPDI	Mean	SD		Mean	SD	Mean	SD
$\ell = 0.334$	Cons	0.95	0.94	0.847	0.009	0.009	0.043	0.007	0.042	0.007
	unCons	0.98	0.99	0.831	0.010	0.011	0.067	0.006	0.065	0.005
$\ell = 0.434$	Cons	0.93	0.94	0.846	0.010	0.010	0.042	0.008	0.042	0.008
	unCons	0.98	0.99	0.832	0.010	0.010	0.063	0.006	0.062	0.005
$\ell = 0.831$	Cons	0.94	0.94	0.844	0.010	0.008	0.043	0.008	0.042	0.007
	unCons	0.97	0.98	0.833	0.009	0.009	0.055	0.006	0.054	0.006
$\ell = 1.443$	Cons	0.95	0.93	0.843	0.011	0.009	0.043	0.008	0.043	0.008
	unCons	0.95	0.94	0.833	0.008	0.009	0.048	0.006	0.047	0.006

"SD" represents the standard deviation across 100 replicates. The column of $\widehat{\mathbb{E}}(\lvert\widehat{r}_p - r_p\rvert)$ gives an estimate of $\mathbb{E}(\lvert\widehat{r}_p - r_p\rvert)$. " 95% CI" and " 95% HPDI" represent the symmetric 95% credible interval and 95% highest posterior density interval respectively

ple sizes, the overall complexity remains substantially in our favor for large values of N.

At present, there are efforts ongoing to collect more data to entirely resolve the proton puzzle. We envision our computational advancements over Zhou et al. (2019) to help analyze the ensuing larger datasets. The computational efficiency also permits us to perform large-scale simulations on a host of test functions developed by physicists where the ground truth is known. The results of such an analysis will be reported elsewhere.

5 Pseudo-data analysis

In this section, we perform a replicated simulation study mimicking the proton data analysis to compare the frequentist operating characteristics of our constrained approach with the corresponding unconstrained one. In addition to the quality of point estimation, we are also interested in the frequentist coverage and lengths of the symmetric and the highest posterior density (HPD) credible intervals for the radius for either approach.

We used a monopole function

$$G_E(Q^2) = \left(1 + \frac{r_p^2\, Q^2}{6}\right)^{-1}$$

as the true electric form factor. The monopole function (Borkowski et al. 1975; Yan et al. 2018) is a popular parametric model for the electric form factor and satisfies all the aforementioned constraints. The parameter r_p plays the role of the radius, whose true value we set to 0.84; a natural choice given the analysis and background in the previous section. We generated the momentum transfer values Q^2 uniformly between $Q^2_{\min} = 0.099$ and $Q^2_{\max} = 1.36$ obtained from the Mainz dataset, and set $x = Q^2/Q^2_{\max} \in [0, 1]$ to be the dimensionless covariate in (2). Setting a lower bound on the Q^2 ensures that we do not get observations too close to the origin, maintaining a similar difficulty in recovering the radius as in the Mainz dataset. To obtain noisy observations y on G_E, we added mean-zero Gaussian noise with $\sigma = 0.01$. The sample size n was set to 500 and we considered 100 simulation replicates.

We fit the constrained model with the set of constraints (11) and prior specification exactly as in the previous section, using the exact version of our algorithm. As a competitor, we also considered an unconstrained version of our method where (ξ^*, ξ) are unrestricted and assigned a MVN prior $\mathcal{N}(0, \tau^2\,\Omega)$ instead of the tMVN prior. We fixed $\xi_0 = 1$ for both the methods. The unconstrained method can be considered (a finite-rank approximation to) a GP prior on f''. We considered two different choices of $N \in \{\lceil n/8 \rceil, \lceil n/4 \rceil\}$ and four different choices of the length-scale parameter $\ell \in \{0.334, 0.434, 0.831, 1.443\}$. These values of ℓ correspond to four correlation values $\{0.05, 0.10, 0.30, 0.50\}$ between the two farthest knots, respectively.

We report a summary of our findings across the 100 replicates regarding the point and interval estimates of r_p in Tables 2 and 3. These were obtained based on 8000 MCMC iterations with a burn-in of 2000 and every third sample post burn-in saved as posterior samples. The HPD intervals were computed from the posterior samples for r_p using **R** package "`HDInterval`". Across either method and all simulation settings, little to no difference was seen between the symmetric 95% credible interval and the 95% HPD interval, suggesting a symmetric shape of the posterior of r_p.

The posterior mean $\widehat{r}_p$ for r_p from the constrained method is uniformly seen to be at least as good as that from the unconstrained method in terms of the mean absolute error risk $\mathbb{E}\lvert\widehat{r}_p - r_p\rvert$ reported in the tables. This is also clearly confirmed by Fig. 3, where we present boxplots of the poste-

Table 3 Results corresponding to $n = 500$ and $N = \lceil n/4 \rceil = 125$ for different choices of ℓ using the exact version of our algorithm (abbreviated as "*Cons*") and the unconstrained counterpart of our method (abbreviated as "*unCons*")

$N = 125$		Coverage Probability		Estimated radius		$\widehat{\mathbb{E}}(\lvert\widehat{r}_p - r_p\rvert)$	Length of 95% CI		Length 95% HPDI	
		95% CI	95% HPDI	Mean	SD		Mean	SD	Mean	SD
$\ell = 0.334$	Cons	0.97	0.97	0.845	0.008	0.007	0.039	0.008	0.038	0.008
	unCons	0.99	1.00	0.832	0.010	0.010	0.074	0.006	0.072	0.006
$\ell = 0.434$	Cons	0.97	0.97	0.843	0.008	0.007	0.037	0.007	0.037	0.006
	unCons	0.99	1.00	0.832	0.010	0.010	0.069	0.007	0.068	0.006
$\ell = 0.831$	Cons	0.99	0.98	0.842	0.008	0.006	0.039	0.007	0.038	0.007
	unCons	0.98	0.99	0.833	0.009	0.009	0.058	0.007	0.057	0.006
$\ell = 1.443$	Cons	0.97	0.97	0.838	0.008	0.006	0.039	0.006	0.038	0.006
	unCons	0.98	0.96	0.834	0.009	0.008	0.051	0.006	0.050	0.005

"SD" represents the standard deviation across 100 replicates. The column of $\widehat{\mathbb{E}}(\lvert\widehat{r}_p - r_p\rvert)$ gives an estimate of $\mathbb{E}(\lvert\widehat{r}_p - r_p\rvert)$. " 95% CI" and " 95% HPDI" represent the symmetric 95% credible interval and 95% highest posterior density interval respectively

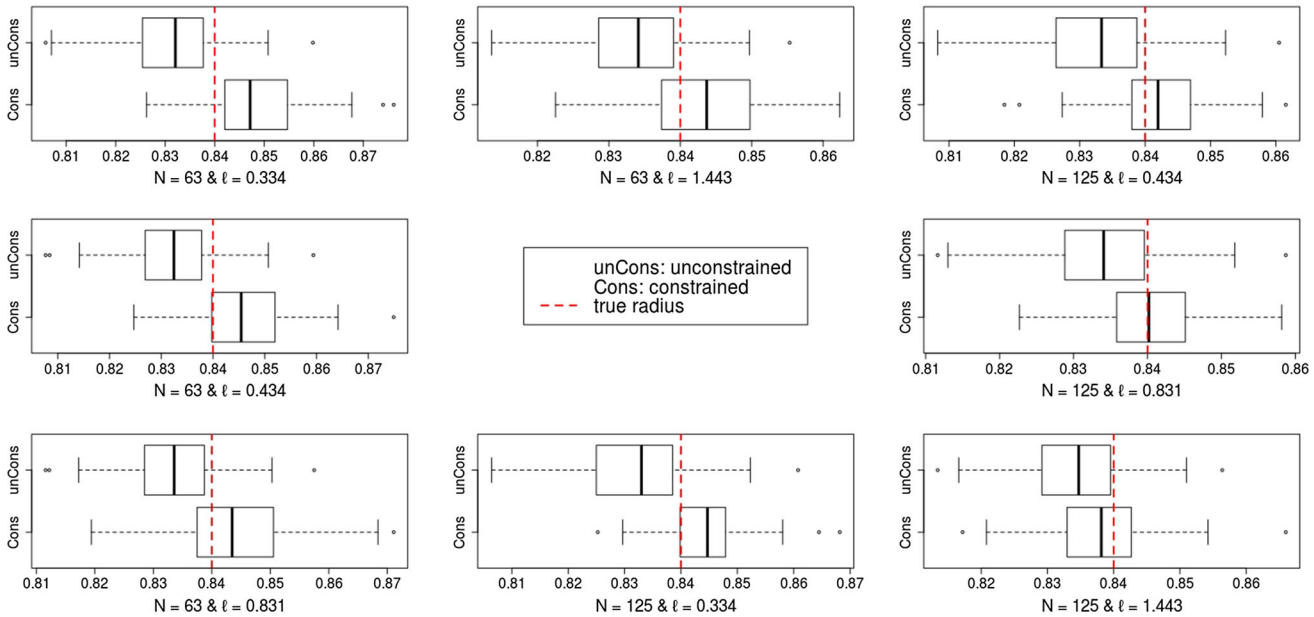


Fig. 3 Boxplots of $\widehat{r}_p$ obtained from both constrained and unconstrained methods corresponding to different combinations of N and ℓ. The true r_p is represented by dashed red line. (Color figure online)

rior means across the 100 replicates for either methods across the 8 different hyperparameter choices. It is also evident from the tables that the constrained method provides more precise uncertainty quantification, measured in terms of the lengths of the credible interval, while maintaining close to nominal coverage in all cases. The unconstrained method tends to over-cover, which may contribute to the longer intervals, which are in some cases more than double the length of that from the constrained method. This simulation exercise thus reinforces the importance of exploiting structural constraints for inferential purpose.

5.1 Cost per-iteration

Our attempts to compare the run-times of our algorithm with the "`tmg`" sampler as in Sect. 3.3 ran into difficulties for this simulation setup as the "`tmg`" sampler often failed to produce answers. One possible reason behind this may be the more complicated form of the constrained region (11) for (ξ^*, ξ), compared to the one in Sect. 3.3. Although not shown here, but based on the cases where "`tmg`" was able to run and produce answers, the per-iteration costs of the exact version of our algorithm was significantly lower compared to the HMC sampler.

6 Discussion

In this article, we have developed a computationally efficient algorithm for constrained Gaussian regression problems. Our approach builds on the promising modeling framework introduced by Maatouk and Bay (2017), which was later adapted by Zhou et al. (2019) to include multiple constraints. An important distinguishing feature of our approach is to begin with a Gaussian process on the first or second derivative of the function depending on the nature of the constraints, which results in a more amenable structure of the prior covariance matrix, facilitating prior and posterior simulations. Both the approximate and exact versions of our algorithms are shown to produce orders-of-magnitude speed-ups over competing Gibbs samplers using state-of-the-art HMC samplers (Pakman and Paninski 2014) for sampling truncated multivariate normals.

For functions of one variable, one may justifiably question the impact of the constraints in recovering the function given a sizable amount of data. Indeed, if a global metric such as prediction loss is considered, the improvement over an unconstrained nonparametric approach (e.g., GP regression) can be minimal for moderate to large sample sizes. However, our simulation study in Sect. 5 reveals that the constraints play a major role in delivering more precise inference for functionals related to higher derivatives of the function. This doesn't seem to be broadly recognized, and it would be quite interesting to explore this from a theoretical standpoint.

For the approximate version of the algorithm, the tuning parameter η should be chosen carefully. Theoretically, higher the value of η, better is the approximation. However, very large values of η may lead to inaccurate results due to numerical issues. Based on the numerous simulation studies we have conducted, but not reported here, $\eta = 50$ gives very accurate results. We used this value for all our simulations and real data studies and recommend it to be a default choice for η. We also note that $\eta < 10$ is not accurate enough, and we suggest exercising caution against using values orders of magnitude larger than 50, which may cause numerical issues.

Future work will focus on using various model selection criterion (such as BIC or its more recent variants such as WBIC or SBIC) to select the number of knots N and the length-scale parameter ℓ, which jointly control the smoothness of the sample paths. Based on numerous additional experiments not reported here, we find that the choice of N generally has a bigger impact than that of ℓ. One should choose N to be large enough to provide an accurate finite dimensional approximation to the parent GP, and at the same time avoid overfitting by choosing too large an N. For example, in the synthetic data analysis, choosing N close to the sample size n showed evidence of overfitting. However, this wasn't the case for the real data analysis. A likely explanation is that the monopole function employed in the simulations can be accurately characterized by a relatively small number of basis functions due to its simple parametric form. In general, there is typically a large range of N values which meet the above criterion. Once N is suitably chosen, we find that a wide range of values of ℓ give desirable answers.

Another interesting direction for future exploration is the choice of the basis. The specific basis employed here was motivated by the theoretical results in Maatouk and Bay (2017) and Zhou et al. (2019) showing an equivalent representation of various function constraints in terms of linear constraints on the coefficients. While our algorithm is trivially adapted to other basis, such an equivalent representation needs to be verified in a case-by-case manner at this point, and a more general theory guaranteeing so (or lack thereof) would be useful.

Acknowledgements We thank Pablo Giulani for sharing the proton dataset and many insightful discussions. We also thank Shuang Zhou for sharing her R code for our comparison in Sect. 4.

Funding Funding was provided by National Science Foundation (Grant Nos. DMS 1613156, DMS 1653404).

A Appendices

A.1 Full conditionals

Consider model (9) and the prior specified in Sect. 3.1. The joint distribution is given by:

$$\pi(Y, \xi_0, \xi, \sigma^2, \tau^2) \propto \left(\sigma^2\right)^{-\frac{n}{2}-1} \exp\left\{-\frac{1}{2\sigma^2}\|Y - \xi_0 1_n - \Psi\xi\|^2\right\} \left(\tau^2\right)^{-\frac{N+1}{2}-1} \exp\left\{-\frac{1}{2\tau^2}\xi^{\mathrm{T}} K^{-1}\xi\right\} \mathbb{1}_{\mathcal{C}_\xi}(\xi)$$

Then,

$\xi \mid Y, \xi_0, \sigma^2, \tau^2$ is truncated multivariate Gaussian truncated on $\mathbb{1}_{\mathcal{C}_\xi}(\xi)$.

$\xi_0 \mid Y, \xi, \sigma^2, \tau^2 \sim \mathcal{N}(\bar{Y}^*, \sigma^2/n)$, where, $\bar{Y}^*$ is average of components of $Y^* = Y - \Psi\xi$.

$\sigma^2 \mid Y, \xi_0, \xi, \tau^2 \sim \mathcal{IG}\big(n/2, \|Y - \xi_0 1_n - \Psi\xi\|^2/2\big)$

$\tau^2 \mid Y, \xi_0, \xi, \sigma^2 \sim \mathcal{IG}\big((N+1)/2, \xi^{\mathrm{T}} K^{-1}\xi/2\big)$

Again, consider model (10) and the prior specified in Sect. 3.2. The joint distribution is given by:

$$\pi(Y, \xi_0, \xi_*, \xi, \sigma^2, \tau^2) \propto \left(\sigma^2\right)^{-\frac{n}{2}-1} \exp\left\{-\frac{1}{2\sigma^2}\|Y - \xi_0 1_n - \xi_* X - \Phi\xi\|^2\right\} \left(\tau^2\right)^{-\frac{N+1}{2}-1} \exp\left\{-\frac{1}{2\tau^2}\xi^{\mathrm{T}} K^{-1}\xi\right\} \mathbb{1}_{\mathcal{C}_\xi}(\xi)$$

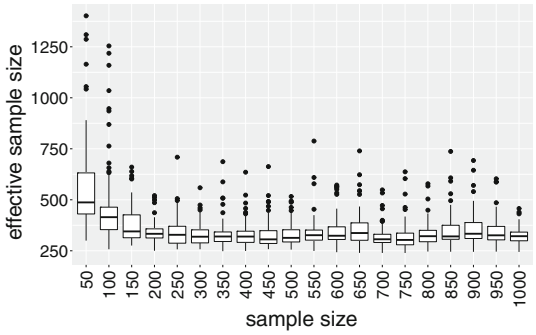


Fig. 4 Boxplots of effective sample sizes of the estimated function value at 75 different points for the monotone function estimation example. The effective sample sizes are calculated based on 10,000 MCMC runs and averaged over 5 random starting points

Then,

$\xi \mid Y, \xi_0, \xi_*, \sigma^2, \tau^2$ is truncated multivariate Gaussian truncated on $\mathbb{1}_{\mathcal{C}_\xi}(\xi)$.

$\xi_0 \mid Y, \xi_*, \xi, \sigma^2, \tau^2 \sim \mathcal{N}(\bar{Y}^*, \sigma^2/n)$, $\bar{Y}^*$ is average of components of $Y^* = Y - \xi_* X - \Phi\xi$.

$\xi_* \mid Y, \xi_0, \xi, \sigma^2, \tau^2 \sim \mathcal{N}(\sum_{i=1}^n x_i y_i^{**} / \sum_{i=1}^n x_i^2, \sigma^2 / \sum_{i=1}^n x_i^2)$, where $Y^{**} = Y - \xi_0 1_n - \Phi\xi$.

$\sigma^2 \mid Y, \xi_0, \xi_*, \xi, \tau^2 \sim \mathcal{IG}\big(n/2, \|Y - \xi_0 1_n - \xi_0 X - \Phi\xi\|^2/2\big)$

$\tau^2 \mid Y, \xi_0, \xi_*, \xi, \sigma^2 \sim \mathcal{IG}\big((N+1)/2, \xi^{\mathrm{T}} K^{-1} \xi/2\big)$

Algorithm 1 was used to draw samples from the full conditional distribution of ξ while sampling from the full conditionals of ξ_0, ξ_*, σ^2 and τ^2 are routine.

A.2 Effective sample sizes for the monotone example in Sect. 3.3

We provide some evidence towards the mixing behavior of our Gibbs sampler by computing the effective sample size of the estimated function value at 75 different test points. The effective sample size is a measure of the amount of the autocorrelation in a Markov chain, and essentially amounts to the number of independent samples in the MCMC path. From an algorithmic robustness perspective, it is desirable that the effective sample sizes remain stable across increasing sample size and/or dimension, and this is the aspect we wish to investigate here. We only report results for the monotonicity constraint; similar behavior is seen for the convexity constraint as well.

We consider 20 different values for the sample size n with equal spacing between 50 and 1000. Note that the dimension of ξ itself grows between 25 and 500 as a result. For each value of n, we run the Gibbs sampler for 12,000 iterations with 5 randomly chosen initializations. For each starting point, we record the effective sample size at each of the 75 test points after discarding the first 2,000 iterations as burn-in, and average them over the different initializations. Figure 4 shows boxplots of these averaged effective sample sizes across n which are seen to be quite stable across growing n.

A.3 R code

We used **R** for the implementation of Algorithm 1 and Durbin's recursion to find the inverse of the Cholesky factor, with the computation of the inverse Cholesky factor optimized with **Rcpp**. We provide our code for implementing the monotone and convex function estimation procedures in Sects. 3.1 and 3.2 in the Github page mentioned in Sect. 1. There are six different functions to perform the MCMC sampling for monotone increasing, monotone decreasing, and convex increasing functions with and without hyperparameter updates. Each of these main functions take x and y as inputs along with other available options, and return posterior samples on ξ_0, ξ^*, ξ, σ, τ and f along with posterior mean and symmetric 95% credible interval of f on a user-specified grid. A detailed description on the available input and output options for each function can be found within the function files.

References

Bardenet, R., Doucet, A., Holmes, C.: On Markov chain Monte Carlo methods for tall data. J. Mach. Learn. Res. **18**(1), 1515–1557 (2017)

Bernauer, J.C, A1 Collaboration: High-precision determination of the electric and magnetic form factors of the proton. In: AIP Conference Proceedings, vol. 1388. AIP, pp. 128–134 (2011)

Bernauer, J.C., Pohl, R.: The proton radius problem. Sci. Am. **310**(2), 32–39 (2014)

Bernauer, J.C., Achenbach, P., Ayerbe Gayoso, C., Böhm, R., Bosnar, D., Debenjak, L., Distler, M.O., Doria, L., Esser, A., Fonvieille, H., et al.: Bernauer et al. reply. Phys. Rev. Lett. **107**(11), 119102 (2011)

Bernauer, J.C., Distler, M.O., Friedrich, J., Walcher, T., Achenbach, P., Ayerbe Gayoso, C., Böhm, R., Bosnar, D., Debenjak, L., Doria, L., et al.: Electric and magnetic form factors of the proton. Phys. Rev. C **90**(1), 015206 (2014)

Borkowski, F., Simon, G.G., Walther, V.H., Wendling, R.D.: On the determination of the proton rms-radius from electron scattering data. Zeitschrift für Physik A Atoms and Nuclei **275**(1), 29–31 (1975)

Bornkamp, B., Ickstadt, K.: Bayesian nonparametric estimation of continuous monotone functions with applications to dose–response analysis. Biometrics **65**(1), 198–205 (2009)

Botev, Z.I.: The normal law under linear restrictions: simulation and estimation via minimax tilting. J. R. Stat. Soc. Ser. B (Stat. Methodol.) **79**(1), 125–148 (2017)

Brezger, A., Steiner, W.J.: Monotonic regression based on Bayesian p-splines: an application to estimating price response functions from store-level scanner data. J. Bus. Econ. Stat. **26**(1), 90–104 (2008)

Cai, B., Dunson, D.B.: Bayesian multivariate isotonic regression splines: applications to carcinogenicity studies. J. Am. Stat. Assoc. **102**(480), 1158–1171 (2007)

Carlson, C.E.: The proton radius puzzle. Prog. Part. Nucl. Phys. **82**, 59–77 (2015)

Chen, H., Yao, D.D.: Dynamic scheduling of a multiclass fluid network. Oper. Res. **41**(6), 1104–1115 (1993)
Cotter, S.L., Roberts, G.O., Stuart, A.M., White, D.: MCMC methods for functions: modifying old algorithms to make them faster. Stat. Sci. **28**(3), 424–446 (2013)
Curtis, S.M., Ghosh, S.K.: A variable selection approach to monotonic regression with Bernstein polynomials. J. Appl. Stat. **38**(5), 961–976 (2011)
Damien, P., Walker, S.G.: Sampling truncated normal, beta, and gamma densities. J. Comput. Graph. Stat. **10**(2), 206–215 (2001)
Geweke, J.: Efficient simulation from the multivariate normal and Student-t distributions subject to linear constraints and the evaluation of constraint probabilities. In: Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface, pp. 571–578. Interface Foundation of North America, Inc., Fairfax, Virginia (1991)
Goldenshluger, A., Zeevi, A.: Recovering convex boundaries from blurred and noisy observations. Ann. Stat. **34**(3), 1375–1394 (2006)
Golub, G.H., van Loan, C.F.: Matrix Computations, 3rd edn. Johns Hopkins University Press, Baltimore (1996)
Higinbotham, D.W., Kabir, A.A., Lin, V., Meekins, D., Norum, B., Sawatzky, B.: Proton radius from electron scattering data. Phys. Rev. C **93**(5), 055207 (2016)
Johndrow, J.E, Orenstein, P., Bhattacharya, A.: Scalable MCMC for Bayes shrinkage priors. arXiv preprint arXiv:1705.00841 (2017)
Kelly, C., Rice, J.: Monotone smoothing with application to dose–response curves and the assessment of synergism. Biometrics **46**(4), 1071–1085 (1990)
Kotecha, J.H., Djuric, P.M.: Gibbs sampling approach for generation of truncated multivariate Gaussian random variables. In: The Proceedings of the International Conference on Acoustics, Speech, and Signal Processing. IEEE (1999)
Lele, A.S., Kulkarni, S.R., Willsky, A.S.: Convex-polygon estimation from support-line measurements and applications to target reconstruction from laser-radar data. J. Opt. Soc. Am. A **9**(10), 1693–1714 (1992)
Lin, L., Dunson, D.B.: Bayesian monotone regression using Gaussian process projection. Biometrika **101**(2), 303–317 (2014)
Maatouk, H., Bay, X.: Gaussian process emulators for computer experiments with inequality constraints. Math. Geosci. **49**(5), 557–582 (2017)
Meyer, R.F., Pratt, J.W.: The consistent assessment and fairing of preference functions. IEEE Trans. Syst. Sci. Cybern. **4**(3), 270–278 (1968)
Meyer, M.C., Hackstadt, A.J., Hoeting, J.A.: Bayesian estimation and inference for generalised partial linear models using shape-restricted splines. J. Nonparametr. Stat. **23**(4), 867–884 (2011)
Murray, I., Prescott Adams, R., MacKay, D.J.C.: Elliptical slice sampling. J. Mach. Learn. Res. W&CP **9**, 541–548 (2010)
Neal, R.M.: Regression and classification using Gaussian process priors. In: Bernardo, J.M. et al. (eds.) Bayesian statistics, vol. 6, pp. 475–501 (1999)
Neelon, B., Dunson, D.B.: Bayesian isotonic regression and trend analysis. Biometrics **60**(2), 398–406 (2004)
Nicosia, G., Rinaudo, S., Sciacca, E.: An evolutionary algorithm-based approach to robust analog circuit design using constrained multi-objective optimization. Knowl. Based Syst. **21**(3), 175–183 (2008)
Pakman, A., Paninski, L.: Exact Hamiltonian Monte Carlo for truncated multivariate Gaussians. J. Comput. Graph. Stat. **23**(2), 518–542 (2014)
Pohl, R., Antognini, A., Nez, F., Amaro, F.D., Biraben, F., Cardoso, J.M.R., Covita, D.S., Dax, A., Dhawan, S., Fernandes, L.M.P., et al.: The size of the proton. Nature **466**(7303), 213–216 (2010)
Prince, J.L., Willsky, A.S.: Constrained sinogram restoration for limited-angle tomography. Opt. Eng. **29**(5), 535–545 (1990)
Reboul, L.: Estimation of a function under shape restrictions. Applications to reliability. Ann. Stat. **33**(3), 1330–1356 (2005)
Riihimäki, J., Vehtari, A.: Gaussian processes with monotonicity information. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, pp. 645–652 (2010)
Rodriguez-Yam, G., Davis, R.A., Scharf, L.L.: Efficient Gibbs sampling of truncated multivariate normal with application to constrained linear regression. Technical Report, Department of Statistics, Columbia University (2004)
Shively, T.S., Walker, S.G., Damien, P.: Nonparametric function estimation subject to monotonicity, convexity and other shape constraints. J. Econom. **161**(2), 166–181 (2011)
Wood, A.T.A., Chan, G.: Simulation of stationary Gaussian processes in $[0, 1]^d$. J. Comput. Graph. Stat. **3**(4), 409–432 (1994)
Yan, X., Higinbotham, D.W., Dutta, D., Gao, H., Gasparian, A., Khandaker, M.A., Liyanage, N., Pasyuk, E., Peng, C., Xiong, W.: Robust extraction of the proton charge radius from electron–proton scattering data. Phys. Rev. C **98**(2), 025204 (2018)
Zhou, S., Giulani, P., Piekarewicz, J., Bhattacharya, A., Pati, D.: Reexamining the proton-radius problem using constrained Gaussian processes. Phys. Rev. C **99**(5), 055202 (2019)