Bayesian linear models for large datasets: Markov chain Monte Carlo or Matheron's update rule

Hassan Maatouk, Didier Rullière, Xavier Bay

Bayesian linear models for large datasets: Markov chain Monte Carlo or Matheron's update rule

Hassan Maatouk[1], Didier Rullière[2], and Xavier Bay[2]

[1] LAMPS, Université de Perpignan via Domitia, 52 av. Paul Alduy, 66860 Cedex 9 Perpignan, France,
hassan.maatouk@univ-perp.fr,
[2] Mines Saint-Étienne, Univ Clermont Auvergne, CNRS, UMR 6158 LIMOS, Institut Henri Fayol, Saint-Étienne, F-42023, France

Abstract. In this paper, we consider Bayesian linear models for large datasets. We discuss two distinct strategies for generating Bayesian linear models with a large number of observations. The first approach employs an efficient Markov chain Monte Carlo (MCMC) method, while the second approach is exact and is based on a modification of Matheron's update rule (MUR) using Bayes' rule. We prove that MUR can be adapted for a large number of observations, resulting in a significant reduction in computational cost. The main advantage of these approaches is that sampling is performed before conditioning rather than after. This allows for the use of highly efficient samplers to generate the prior Gaussian vector when the precision covariance matrix exhibits special structures, such as Toeplitz, block-Toeplitz or sparsity. An empirical comparison between these two efficient approaches in terms of computational running time and prediction accuracy is conducted using both synthetic and real-world data studies.

Keywords: Bayesian linear models, large datasets, Matheron's update rule, Elliptical Slice Sampling, MCMC, Toeplitz

1 Introduction

Gaussian Processes (GPs) have become a popular choice in Bayesian approaches for nonparametric function estimation due to their flexibility, probabilistic nature, and ability to model uncertainty effectively [1, 22, 26]. Unlike parametric methods, which assume a fixed functional form with a finite number of parameters, GPs provide a distribution over functions, allowing for infinite-dimensional flexibility. This makes GPs particularly well-suited for scenarios where the true functional form is unknown or highly complex.

Several finite-dimensional Bayesian linear models have been developed in the literature. Examples include the truncated Karhunen-Loève expansion (KLE) [14, 25], the B-spline expansion [9], Bernstein polynomials [2, 3, 6], and the compactly supported basis expansion [15, 16]. These approaches are described as the *weight-space view* of GPs in [26, Sect. 2.1]. The primary advantages of these methods lie in their simplicity of implementation and interpretability.

In this paper, we develop two distinct general methods for generating finite-dimensional Bayesian linear models with a large number of observations. The first approach is based

on Markov chain Monte Carlo (MCMC), while the second leverages a modification of Matheron's update rule (MUR) [12, 13] using Bayes' rule. The main advantage of these approaches is that sampling is performed before conditioning rather than after. Consequently, if the prior covariance matrix exhibits special structures, such as Toeplitz or block-Toeplitz (i.e., when evaluating stationary product kernels on a regularly spaced grid) [28, 29] or banded and sparse forms [8], highly efficient samplers can be employed. Examples include the Fast Fourier Transform (FFT) [28] and the fast large-scale approaches developed in [18–20]. For example, the authors in [4] used MUR to propose a fast algorithm for simulating a *hyperplane-truncated* multivariate normal (MVN) distribution, where the prior covariance (or precision) matrix can be expressed as a positive-definite matrix minus (or plus) a low-rank symmetric matrix.

This paper is organized as follows. In Sect. 2, we briefly review Bayesian nonparametric function estimation. Section 3 is devoted to the finite-dimensional Bayesian linear models, where two methods for handling large datasets are developed. The excellent performance of these two approaches is evaluated through both synthetic and real-world data studies in Sects. 4 and 5, respectively.

2 Bayesian nonparameteric function estimation

In this section, the statistical problem of recovering an unknown function $f : \mathcal{X} \subset \mathbb{R}^d \longrightarrow \mathbb{R}$ from the training samples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ with Gaussian noise is considered

$$y_i = f(\boldsymbol{x}_i) + \epsilon_i, \quad i = 1, \ldots, n. \tag{1}$$

We assume that the noises $\{\epsilon_i\}$ are i.i.d. zero-mean Gaussian with constant variance σ^2 and are independent of the input vectors (covariates) $\boldsymbol{x}_i \in \mathbb{R}^d$. The n responses $\{y_i\}$, together with the covariates, form the training samples $\{(\boldsymbol{x}_i, y_i)\}$. The unknown function $f : \mathcal{X} \subset \mathbb{R}^d \longrightarrow \mathbb{R}$ is the target function generating the data $\boldsymbol{y} = [y_1, \ldots, y_n]^\top \in \mathbb{R}^n$ using (1). The set $\mathcal{X}$ is a compact subset of $\mathbb{R}^d$. Without loss of generality, we assume $\mathcal{X}$ to be the unit hypercube.

2.1 Gaussian processes

In the Bayesian framework, a GP is defined as a stochastic process where any finite collection of its random variables follows a MVN distribution. This property characterizes GPs as infinite-dimensional generalizations of MVN distributions. A GP is completely characterized by two functions: its mean function $m(\cdot)$ and its covariance function $k(\cdot, \cdot)$. The covariance function, also known as the kernel, is particularly important as it determines the smoothness properties of the sample paths generated by the GP. In this paper, the stationary Matérn family of covariance functions [10] is used, which in one-dimension is defined as follows

$$k(x, x') := \tau^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}}{\ell} |x - x'| \right)^{\nu} B_\nu \left(\frac{\sqrt{2\nu}}{\ell} |x - x'| \right), \tag{2}$$

for any $x, x' \in \mathcal{X}$, where $\Gamma(\cdot)$ is the Gamma function and $B_\nu(\cdot)$ denotes the modified Bessel function of the second kind of order ν. The positive kernel parameter τ^2 is

referred to as *signal variance*. It is worth noting that a process with the Matérn kernel of order ν admits sample paths that are $\lceil \nu - 1 \rceil$ times differentiable [26, Sect. 4.2.1].

Suppose that this GP is denoted by Z, then we can write

$$(Z(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}} \sim \mathcal{GP}(m(\cdot), k(\cdot,\cdot)), \quad \boldsymbol{x} \in \mathcal{X}.$$

We define the mean function $m(\boldsymbol{x})$ and the covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$ of $(Z(\boldsymbol{x}))$ as

$$\begin{aligned} &m(\boldsymbol{x}) = \mathrm{E}[Z(\boldsymbol{x})], \quad \forall \boldsymbol{x} \in \mathcal{X}; \\ &k(\boldsymbol{x}, \boldsymbol{x}') = \mathrm{E}\left[(Z(\boldsymbol{x}) - m(\boldsymbol{x}))(Z(\boldsymbol{x}') - m(\boldsymbol{x}'))\right], \quad \forall \boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}. \end{aligned}$$

Furthermore, Z can be decomposed as its mean function and a zero-mean GP:

$$Z(\boldsymbol{x}) = m(\boldsymbol{x}) + Y(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathcal{X},$$

where $(Y(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}}$ is a zero-mean GP with covariance function $k(\cdot,\cdot)$.

2.2 Gaussian processes conditionally on data

In the regression framework (1), GPs are known as powerful prior distributions over functions of one or more input variables [22, 26]. Let us first denote the design matrix by $\mathbb{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]^\top \in \mathbb{R}^{n\times d}$, which is obtained by aggregated the input vectors $\{\boldsymbol{x}_i\}$. The set of noisy observations is given by $\{Y(\mathbb{X}) + \boldsymbol{\epsilon} = \boldsymbol{y}\}$, where $\boldsymbol{\epsilon} = [\epsilon_1, \ldots, \epsilon_n]^\top$. According to [26, Appendix A.2], conditioning the GP prior $(Y(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}}$ on the observations, we obtain a GP

$$\{Y(\cdot) \text{ s.t. } Y(\mathbb{X}) + \boldsymbol{\epsilon} = \boldsymbol{y}\} \sim \mathcal{GP}\left(\widetilde{m}(\cdot), \widetilde{k}(\cdot,\cdot)\right).$$

The posterior mean function $\widetilde{m}(\cdot)$ and covariance function $\widetilde{k}(\cdot,\cdot)$ are given by

$$\begin{cases} \widetilde{m}(\boldsymbol{x}) = \mathrm{E}\left[Y(\boldsymbol{x}) | Y(\mathbb{X}) + \boldsymbol{\epsilon} = \boldsymbol{y}\right] = k(\boldsymbol{x}, \mathbb{X})^\top \left(k(\mathbb{X}, \mathbb{X}) + \sigma^2 \boldsymbol{I}_n\right)^{-1} \boldsymbol{y}; \\ \widetilde{k}(\boldsymbol{x}, \boldsymbol{x}') = k(\boldsymbol{x}, \boldsymbol{x}') - k(\boldsymbol{x}, \mathbb{X})^\top (k(\mathbb{X}, \mathbb{X}) + \sigma^2 \boldsymbol{I}_n)^{-1} k(\boldsymbol{x}', \mathbb{X}); \end{cases} \tag{3}$$

with $\boldsymbol{I}_n$ the $n \times n$ identity matrix and $k(\boldsymbol{x}, \mathbb{X})$ the vector of covariance between $Y(\boldsymbol{x})$ and $Y(\mathbb{X})$, i.e., $k(\boldsymbol{x}, \mathbb{X}) = [k(\boldsymbol{x}, \boldsymbol{x}_1), \ldots, k(\boldsymbol{x}, \boldsymbol{x}_n)]^\top$. The formula for $\widetilde{m}$ in (3) is referred to as the prediction with noisy observations. If the standard deviation σ of the noise $\boldsymbol{\epsilon}$ is set to zero, the term *noise-free* observations is used, as in the context of 'computer experiments' [23].

3 Finite-dimensional Bayesian linear models

In this section, we suppose that the GP $(Y(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}}$ is approximated by a finite-dimensional Bayesian linear model as follows:

$$Y(\boldsymbol{x}) \approx \phi(\boldsymbol{x})^\top \boldsymbol{\xi} := Y^N(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathcal{X}, \tag{4}$$

where $\boldsymbol{\xi} \in \mathbb{R}^N$ is a zero-mean Gaussian vector (weight of the model) with a positive-definite covariance matrix $\tau^2\boldsymbol{K}$, i.e., $\boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{0}, \tau^2\boldsymbol{K})$. The basis function $\phi(\cdot)$ is a sequence of deterministic basis functions such that $\phi(\boldsymbol{x}) \in \mathbb{R}^N$, for any $\boldsymbol{x} \in \mathcal{X}$. The finite-dimensional linear model in (4) is known as *weight-space view* of GPs in [26, Sect. 2.1]. The results developed in this section can be applied to several finite-dimensional Bayesian linear models, such as those mentioned in the introduction. For simplicity of notations, we denote by $\boldsymbol{X} = \phi(\mathbb{X}) \in \mathbb{R}^{n\times N}$, where $\mathbb{X}$ is the $n \times d$ design matrix and the i^{th} row $\boldsymbol{X}_{i,} = \phi(\mathbb{X}_{i,}^\top)^\top$. In this case, the set of noisy observations $\{Y^N(\mathbb{X}) + \boldsymbol{\epsilon} = \boldsymbol{y}\}$ can be written in matrix form as follows:

$$\boldsymbol{X}\boldsymbol{\xi} + \boldsymbol{\epsilon} = \boldsymbol{y},$$

where $\boldsymbol{\epsilon} = [\epsilon_1, \ldots, \epsilon_n]^\top$ is a zero-mean Gaussian noise vector with covariance matrix $\sigma^2\boldsymbol{I}_n$. It is worth noting that the vector $Y^N(\mathbb{X})$ follows a Gaussian distribution with zero-mean and covariance matrix $\tau^2\boldsymbol{X}\boldsymbol{K}\boldsymbol{X}^\top$. Conditionally on the observations $\boldsymbol{y}$

$$\{\boldsymbol{\xi}|\boldsymbol{X}\boldsymbol{\xi} + \boldsymbol{\epsilon} = \boldsymbol{y}\} \sim \mathcal{N}\left(\widetilde{\boldsymbol{\mu}}, \widetilde{\boldsymbol{K}}\right), \quad \text{where,} \tag{5}$$

$$\begin{cases} \widetilde{\boldsymbol{\mu}} = \tau^2(\boldsymbol{X}\boldsymbol{K})^\top(\tau^2\boldsymbol{X}\boldsymbol{K}\boldsymbol{X}^\top + \sigma^2\boldsymbol{I}_n)^{-1}\boldsymbol{y}; \\ \widetilde{\boldsymbol{K}} = \tau^2\boldsymbol{K} - \tau^4(\boldsymbol{X}\boldsymbol{K})^\top(\tau^2\boldsymbol{X}\boldsymbol{K}\boldsymbol{X}^\top + \sigma^2\boldsymbol{I}_n)^{-1}\boldsymbol{X}\boldsymbol{K}. \end{cases} \tag{6}$$

Sampling the posterior distribution in (5), involves generating a Gaussian vector of dimension N. When N is large, this approach becomes prohibitive due to its computational complexity of order $\mathcal{O}(N^3)$ [11]. Additionally, the predictive equations in (6) require a matrix inversion of dimension $n \times n$, making this approach infeasible for a large number of observations n.

In the following section, we develop two different approaches for handling this issue.

3.1 First approach: Markov chain Monte Carlo

In this section, we explore highly efficient MCMC methods to address the problem of sampling from a finite-dimensional Bayesian linear model (4) applied to a large number of observations. According to [26, Sect. 2.1.1] and Bayes' rule, we have the following posterior distribution:

$$p(\boldsymbol{\xi}|\boldsymbol{X}, \boldsymbol{y}) := \frac{p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\xi})p(\boldsymbol{\xi})}{p(\boldsymbol{y}|\boldsymbol{X})}, \tag{7}$$

where $p(\boldsymbol{y}|\boldsymbol{X})$ is the normalizing constant, also known as the *marginal likelihood*. It is independent of $\boldsymbol{\xi}$ and given by

$$p(\boldsymbol{y}|\boldsymbol{X}) = \int_{\mathbb{R}^N} p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\xi})p(\boldsymbol{\xi})\mathrm{d}\boldsymbol{\xi}.$$

By developing the likelihood and prior in (7), we obtain

$$\begin{aligned} p(\boldsymbol{\xi}|\boldsymbol{X}, \boldsymbol{y}) &\propto \exp\left(-\frac{1}{2\sigma^2}[\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\xi}]^\top[\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\xi}]\right)\exp\left(-\frac{1}{2\tau^2}\boldsymbol{\xi}^\top\boldsymbol{K}^{-1}\boldsymbol{\xi}\right) \\ &\propto \exp\left(-\frac{1}{2}[\boldsymbol{\xi} - \boldsymbol{\mu}]^\top\boldsymbol{\Sigma}^{-1}[\boldsymbol{\xi} - \boldsymbol{\mu}]\right), \end{aligned} \tag{8}$$

where

$$\begin{cases} \boldsymbol{\mu} = \left[\boldsymbol{X}^\top \boldsymbol{X}/\sigma^2 + \boldsymbol{K}^{-1}/\tau^2\right]^{-1} \boldsymbol{X}^\top \boldsymbol{y}/\sigma^2; \\ \boldsymbol{\Sigma} = \left[\boldsymbol{X}^\top \boldsymbol{X}/\sigma^2 + \boldsymbol{K}^{-1}/\tau^2\right]^{-1}. \end{cases} \tag{9}$$

As in (5), the posterior distribution in (7) is Gaussian and is given by $\{\boldsymbol{\xi}|\boldsymbol{X}, \boldsymbol{y}\} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. It is worth noting that the *mean* of the Gaussian posterior distribution $\{\boldsymbol{\xi}|\boldsymbol{X}, \boldsymbol{y}\}$ is also its mode, which is referred to as the *maximum a posteriori* (MAP) estimate of $\boldsymbol{\xi}$. The predictive equations in (6) and (9) are equivalent, and the two approaches are referred to as *direct* approaches, as sampling is performed after conditioning. However, the predictive equations in (9) require matrix inversion of dimension $N \times N$, where N is the dimension of the Gaussian vector $\boldsymbol{\xi}$. This results in an efficient approach when the dimension N is low and the number of samples n is high.

Now, we explain how MCMC approaches can handle the problem of generating the posterior distribution $\{\boldsymbol{\xi}|\boldsymbol{X}, \boldsymbol{y}\}$ when both N and n are large. The posterior probability density function (pdf) in (8) is proportional to the product of a likelihood function and a zero-mean Gaussian prior:

$$\begin{aligned} p(\boldsymbol{\xi}|\boldsymbol{X}, \boldsymbol{y}) &\propto \underbrace{\exp\left(-\frac{1}{2\sigma^2}[\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\xi}]^\top [\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\xi}]\right)}_{\text{likelihood}} \underbrace{\exp\left(-\frac{1}{2\tau^2}\boldsymbol{\xi}^\top \boldsymbol{K}^{-1}\boldsymbol{\xi}\right)}_{\text{untruncated prior}} \\ &= L(\boldsymbol{\xi})\mathcal{N}(\boldsymbol{\xi}; \boldsymbol{0}, \tau^2\boldsymbol{K}). \end{aligned}$$

The logarithm of the likelihood function $L(\boldsymbol{\xi})$ can be expressed as follows:

$$\log[L(\boldsymbol{\xi})] = -\frac{1}{2\sigma^2}\|y - \boldsymbol{X}\boldsymbol{\xi}\|^2. \tag{10}$$

The logarithm function in (10), which has a computational complexity of order $\mathcal{O}(nN)$ will be evaluated at each MCMC iteration. In this context, sampling from (8) can be performed using Metropolis-Hastings (MH) proposals [5, 22]:

$$\boldsymbol{\xi}' = \rho\boldsymbol{\nu} + \sqrt{1-\rho^2}\boldsymbol{\xi}, \quad \boldsymbol{\nu} \sim \mathcal{N}(\boldsymbol{0}, \tau^2\boldsymbol{K}), \tag{11}$$

where $\rho \in [-1, 1]$ is a step-size parameter, $\boldsymbol{\xi}$ is the current state, and $\boldsymbol{\xi}'$ is the proposal state. Recall that the MH acceptance ratio, $\alpha = \min\left\{1, L(\boldsymbol{\xi}')/L(\boldsymbol{\xi})\right\}$ depends solely on the likelihood ratio and is independent of ρ. Furthermore, this method is straightforward to implement and can be readily applied to a wide range of models with Gaussian priors.

The Elliptical Slice Sampling (ESS) is based on the parametrization $\rho = \sin(\theta)$ in (11), providing an adaptive and automated method for tuning the step-size parameter ρ, which guarantees acceptance at every step. Consequently, the MH proposal in (11) is reformulated as follows:

$$\boldsymbol{\xi}' = \sin(\theta)\boldsymbol{\nu} + \cos(\theta)\boldsymbol{\xi}, \quad \boldsymbol{\nu} \sim \mathcal{N}(\boldsymbol{0}, \tau^2\boldsymbol{K}),$$

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. For a given value of θ, a uniform

random number is generated and compared with the likelihood ratio $L(\boldsymbol{\xi}')/L(\boldsymbol{\xi})$. If the proposal $\boldsymbol{\xi}'$ is rejected, one shrinks the bracket of θ, and continues this process until acceptance. Detailed guidelines for shrinking the bracket are provided in [21].

Unlike *direct* approaches, the primary advantage of the MCMC approach developed in this section is that it avoids matrix inversion. Additionally, the sampling procedure is performed before conditioning rather than after. As a result, highly efficient samplers can be employed when the prior covariance matrix $\boldsymbol{K}$ exhibits special structures, such as Toeplitz, block-Toeplitz or sparsity. Furthermore, the MCMC method introduced in this section is capable of addressing more complex posterior inference (complex likelihood function), such as when additional shape constraints are required [17]. However, this sampling method remains an approximation of the posterior distribution. Moreover, it requires evaluating the log-likelihood function (10) at each MCMC iteration. To address this limitation, the following section introduces an exact sampling method for generating the posterior distribution in (5), specifically tailored for large datasets.

3.2 Second approach: Matheron's update rule

In this section, we develop a new approach based on the MUR methodology. The MUR, which first appeared in geostatistics [12, 13], is an exact method for sampling conditional Gaussian vectors. It was recently explored by [27] in the context of machine learning and by [4] in the field of Bayesian analysis for high-dimensional regression. Let us first recall the following result:

Proposition 1 (Matheron's update rule (MUR)). *Let $\boldsymbol{\xi}$ be an N-dimensional Gaussian vector with a prior distribution characterized by a mean vector $\boldsymbol{\mu}$ and a covariance matrix $\tau^2\boldsymbol{K}$. Suppose that $\boldsymbol{X} \in \mathbb{R}^{n\times N}$ is a given matrix of rank n, and $\boldsymbol{y} \in \mathbb{R}^n$ is an output vector representing the data. Then*

$$\{\boldsymbol{\xi}|\boldsymbol{X}\boldsymbol{\xi}=\boldsymbol{y}\} \overset{d}{=} \underbrace{\boldsymbol{\xi}}_{\textit{prior}} + \underbrace{(\boldsymbol{X}\boldsymbol{K})^\top \left(\boldsymbol{X}\boldsymbol{K}\boldsymbol{X}^\top\right)^{-1}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\xi})}_{\textit{update}}. \tag{12}$$

Additionally, we have

$$\phi(\cdot)^\top\{\boldsymbol{\xi}|\boldsymbol{X}\boldsymbol{\xi}=\boldsymbol{y}\} \overset{d}{=} \phi(\cdot)^\top\left[\boldsymbol{\xi} + (\boldsymbol{X}\boldsymbol{K})^\top \left(\boldsymbol{X}\boldsymbol{K}\boldsymbol{X}^\top\right)^{-1}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\xi})\right],$$

where $\phi(\cdot)$ is the basis vector appearing in the Bayesian linear model (4)*.*

The problem in (12) is known as *hyperplane-truncated* MVN distribution [4]. The proof of Proposition 1 is provided in [19, Appendix B] and in [27, Theorem 1]. More generally, if the data are observed with independent Gaussian noise $\{\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon}=\boldsymbol{y}\}$, where $\boldsymbol{\epsilon}\sim\mathcal{N}(\boldsymbol{0},\sigma^2\boldsymbol{I}_n)$, then

$$\{\boldsymbol{\xi}|\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon}=\boldsymbol{y}\} \overset{d}{=} \boldsymbol{\xi} + \tau^2(\boldsymbol{X}\boldsymbol{K})^\top \left(\tau^2\boldsymbol{X}\boldsymbol{K}\boldsymbol{X}^\top + \sigma^2\boldsymbol{I}_n\right)^{-1}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\xi}-\boldsymbol{\epsilon}). \tag{13}$$

Equation (13) is a simple extension of the result in Proposition 1.

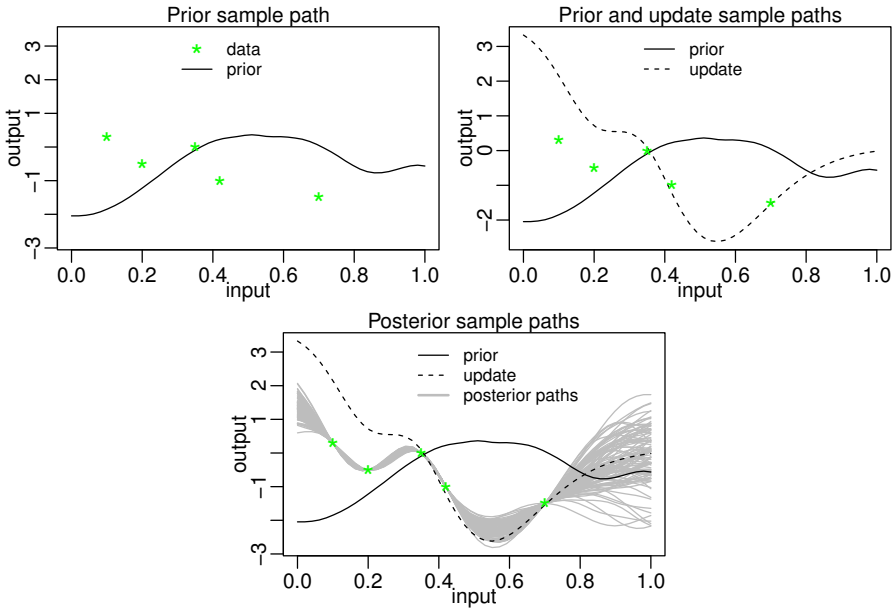

Fig. 1. Visual representation of the MUR (*noise-free* case). Top left: a single path of the prior together with the observations (green stars). Top right: the corresponding update sample path, derived from (12) is displayed as black dashed curve. Bottom: the posterior paths (gray solid curves) are obtained by combining the prior and the update as per (12)

Figure 1 visually represents Proposition 1 using the Bayesian linear model proposed by [15]. The *noise-free* case is considered. In the top left panel, we only illustrate one sample path of the prior with the training samples (green stars). The prior is generated using a zero-mean Gaussian vector, with its covariance matrix derived from the Matérn covariance function (2), specified by a smoothness parameter $\nu = 5/2$ and a length-scale parameter $\ell = 0.3$. In the top left panel, we added the corresponding update sample paths, derived from Equation (12) (black dashed curve). In the bottom panel, we show one hundred posterior sample paths (gray curves) obtained by combining the priors and the updates using Equation (12). It is worth noting that the prior sample path is independent of the data, whereas the update follows the trend of the data. The posterior sample paths (gray curves) effectively interpolate the data (green stars). The main advantage of this method is that sampling is performed before conditioning rather than after. Consequently, if the precision covariance matrix $\boldsymbol{K}$ exhibits a special structure, such as block-Toeplitz, efficient sampling approaches like those mentioned in the introduction can be employed. The MUR has demonstrated high stability compared to eigendecomposition and Cholesky factorization [18]. However, this method still has some limitations, particularly when a large number of observations is required. To overcome this issue, we propose the following adaptation.

Proposition 2 (MUR for a large number of observations). *Under the same settings as Proposition 1, we have*

$$\{\boldsymbol{\xi}|\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon}=\boldsymbol{y}\} \overset{d}{=} \underbrace{\boldsymbol{\xi}}_{\textit{prior}} + \underbrace{(\boldsymbol{X}^\top\boldsymbol{X}/\sigma^2+\boldsymbol{K}^{-1}/\tau^2)^{-1}\boldsymbol{X}^\top(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\xi}-\boldsymbol{\epsilon})/\sigma^2}_{\textit{update}},$$

where $\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\xi}-\boldsymbol{\epsilon}$ represents the residual and σ^2 is the variance of the noise.

Before proving Proposition 2, let us present the following Lemma.

Lemma 1. *Consider three random vectors $\boldsymbol{V}_1 \in \mathbb{R}^N$, $\boldsymbol{V}_2 \in \mathbb{R}^n$ and $\boldsymbol{V}_3 \in \mathbb{R}^N$ s.t.*

$$\boldsymbol{V}_1 \overset{d}{=} f(\boldsymbol{V}_2)+\boldsymbol{V}_3,$$

where f is a measurable function of $\boldsymbol{V}_2$ and where $\boldsymbol{V}_2$ is independent of $\boldsymbol{V}_3$. Then,

$$\{\boldsymbol{V}_1|\boldsymbol{V}_2=\boldsymbol{\theta}\} \overset{d}{=} f(\boldsymbol{\theta})+\boldsymbol{V}_3,$$

for any $\boldsymbol{\theta} \in \mathbb{R}^n$.

Proof. The proof is provided in [27, Lemma 2].

Proof (Proof of Proposition 2). From the equivalent between the two *direct* approaches Equations (6) and (9), we have

$$(\boldsymbol{X}^\top\boldsymbol{X}/\sigma^2+\boldsymbol{K}^{-1}/\tau^2)\boldsymbol{X}^\top/\sigma^2 = \tau^2\boldsymbol{K}\boldsymbol{X}^\top(\tau^2\boldsymbol{X}\boldsymbol{K}\boldsymbol{X}^\top+\sigma^2\boldsymbol{I}_n)^{-1}. \tag{14}$$

Let $\boldsymbol{V}_3 := \boldsymbol{\xi}-(\boldsymbol{X}^\top\boldsymbol{X}/\sigma^2+\boldsymbol{K}^{-1}/\tau^2)^{-1}\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon})/\sigma^2$. Additionally, we have

$$\mathrm{E}[\boldsymbol{\xi}|\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon}] = (\boldsymbol{X}^\top\boldsymbol{X}/\sigma^2+\boldsymbol{K}^{-1}/\tau^2)^{-1}\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon})/\sigma^2.$$

Thus, we can write:

$$\boldsymbol{\xi} = \mathrm{E}[\boldsymbol{\xi}|\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon}]+(\boldsymbol{\xi}-\mathrm{E}[\boldsymbol{\xi}|\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon}]) = (\boldsymbol{X}^\top\boldsymbol{X}/\sigma^2+\boldsymbol{K}^{-1}/\tau^2)^{-1}\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon})/\sigma^2+\boldsymbol{V}_3.$$

Let $\boldsymbol{V}_1=\boldsymbol{\xi}$ and $\boldsymbol{V}_2=\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon}$. Since $\boldsymbol{V}_2$ and $\boldsymbol{V}_3$ are jointly Gaussian but uncorrelated, it follows that they are independent. Indeed,

$$\begin{aligned}
\mathrm{Cov}(\boldsymbol{V}_2,\boldsymbol{V}_3) &= \mathrm{Cov}(\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon}, \boldsymbol{\xi}-(\boldsymbol{X}^\top\boldsymbol{X}/\sigma^2+\boldsymbol{K}^{-1}/\tau^2)^{-1}\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon})/\sigma^2)\\
&= \tau^2\boldsymbol{X}\boldsymbol{K} - \mathrm{Var}(\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon})[(\boldsymbol{X}^\top\boldsymbol{X}/\sigma^2+\boldsymbol{K}^{-1}/\tau^2)^{-1}\boldsymbol{X}^\top/\sigma^2]^\top\\
&= \tau^2\boldsymbol{X}\boldsymbol{K} - (\tau^2\boldsymbol{X}\boldsymbol{K}\boldsymbol{X}^\top+\sigma^2\boldsymbol{I}_n)(\tau^2\boldsymbol{X}\boldsymbol{K}\boldsymbol{X}^\top+\sigma^2\boldsymbol{I}_n)^{-1}\tau^2\boldsymbol{X}\boldsymbol{K}\\
&= \tau^2\boldsymbol{X}\boldsymbol{K}-\tau^2\boldsymbol{X}\boldsymbol{K} = \boldsymbol{0}_{n,N},
\end{aligned}$$

where $\boldsymbol{0}_{n,N}$ is the $n\times N$ zero matrix. The second-to-last line is done using Equation (14). Setting $f(\boldsymbol{V}_2)=(\boldsymbol{X}^\top\boldsymbol{X}/\sigma^2+\boldsymbol{K}^{-1}/\tau^2)^{-1}\boldsymbol{X}^\top\boldsymbol{V}_2/\sigma^2$ and using Lemma 1, we obtain

$$\begin{aligned}
\{\boldsymbol{\xi}|\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon}=\boldsymbol{y}\} &\overset{d}{=} (\boldsymbol{X}^\top\boldsymbol{X}/\sigma^2+\boldsymbol{K}^{-1}/\tau^2)^{-1}\boldsymbol{X}^\top\boldsymbol{y}/\sigma^2+\boldsymbol{\xi}-\\
&\qquad (\boldsymbol{X}^\top\boldsymbol{X}/\sigma^2+\boldsymbol{K}^{-1}/\tau^2)^{-1}\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{\xi}+\boldsymbol{\epsilon})/\sigma^2\\
&\overset{d}{=} \boldsymbol{\xi}+(\boldsymbol{X}^\top\boldsymbol{X}/\sigma^2+\boldsymbol{K}^{-1}/\tau^2)^{-1}\boldsymbol{X}^\top(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\xi}-\boldsymbol{\epsilon})/\sigma^2.
\end{aligned}$$

Hence, the claim follows.

Unlike Proposition 1, the update in Proposition 2 involves a matrix inversion of size $N \times N$. This provides an efficient approach for handling a large number of observations n, provided that N remains reasonably small. As for the MCMC approach, the sampling procedure is carried out prior to conditioning. This involves using highly efficient methods when the dimension of the prior N is large and the prior covariance matrix exhibits special structures, such as Toeplitz, block-Toeplitz or sparsity. The performance of this method is investigated in the context of Bayesian nonparametric function estimation. Notably, the result in Proposition 2 enables an exact sampling method for generating the posterior distribution in (5). As a result, the proposed approach eliminates the need for MCMC sampling and the evaluation of a likelihood function at each MCMC iteration.

4 Performance illustration

The aim of this section is to investigate the performance of the proposed two strategies in terms of computational running time and prediction accuracy. To do this, several Bayesian linear models can be employed such as the truncated Karhunen-Loève expansion (KLE) [14, 25], the B-spline expansion [7, 9] and Bernstein polynomial [2, 3, 6]. In this section, the compact basis expansion approach proposed by [15] is considered. In order for the article to be self-contained, we briefly review this method. If $(Y(x))_{x\in\mathcal{X}}$ is a zero-mean GP with covariance function $k(\cdot,\cdot)$, then it can be approximated by the following finite-dimensional Bayesian linear model:

$$Y^N(x) := \sum_{j=1}^{N} Y(t_j)\phi_j(x) = \sum_{j=1}^{N} \xi_j \phi_j(x), \quad x \in \mathcal{X}, \tag{15}$$

where, we denote $\xi_j = Y(t_j)$ for any $j \in \{1,\ldots,N\}$, with $\{t_j\}$ a sequence of $N \geq 2$ equally-spaced knots, i.e., $0 = t_1 < \ldots < t_N = 1$. Since Y is assumed to be a zero-mean GP, then the basis coefficients $\{\xi_j\}$ form a zero-mean Gaussian vector with covariance matrix $\tau^2\boldsymbol{K}$, i.e., $\boldsymbol{\xi} = [\xi_1,\ldots,\xi_N]^\top \sim \mathcal{N}(\boldsymbol{0}, \tau^2\boldsymbol{K})$. Indeed,

$$\mathrm{Cov}(\xi_i,\xi_j) = \mathrm{Cov}(Y(t_i),Y(t_j)) = k(t_i,t_j) = \tau^2\boldsymbol{K}_{i,j}, \quad \forall i,j = 1,\ldots,N,$$

where $k(\cdot,\cdot)$ is the covariance function of the parent GP Y. The function ϕ_j is the compactly supported basis function associated to the knot $t_j := (j-1)\delta_N$

$$\phi_j(x) := \begin{cases} 1 - \frac{|x-t_j|}{\delta_N} & \text{if } t_j \in [t_{j-1}, t_{j+1}]; \\ 0 & \text{otherwise;} \end{cases}$$

with $\delta_N = 1/(N-1)$. We define $\phi_1(x) = 1 - |x-t_j|/\delta_N$ if $t_j \in [t_1,t_2]$, and zero otherwise. Additionally, we define $\phi_N(x) = 1 - |x-t_j|/\delta_N$ if $t_j \in [t_{N-1},t_N]$, and zero otherwise (B-spline of degree 1 [9, Fig. 1.a]). As proved in [15], $Y^N(\cdot)$ converges uniformly to $Y(\cdot)$ as N tends to infinity (with probability one). This implies that every realization of $Y^N(\cdot)$ converges uniformly to a realization of $Y(\cdot)$.

In Fig. 2, the finite-dimensional Bayesian linear model defined in (15) with $N = 50$ is employed. We consider the target function $f(x) = x\cos(10x)$ (black solid curve).

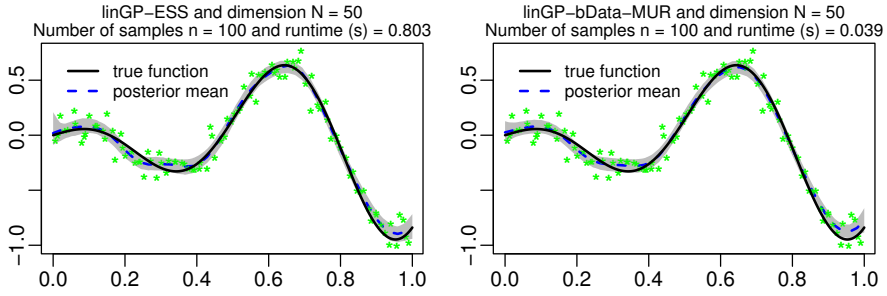

Fig. 2. Performance accuracy of the finite-dimensional Bayesian linear model (4) for $N = 50$. The number of samples (green stars) is fixed at $n = 100$. The gray shaded area represents the 95% confidence interval based on 6,000 sample paths. The highly efficient MCMC approach developed in Sect. 3.1 is employed in the left panel, while the proposed MUR for big data developed in Sect. 3.2 is applied in the right panel

The green stars represent the $n = 100$ training samples generated from (1) using the target function f and a zero-mean Gaussian noise $\mathcal{N}(0, \sigma^2)$, with variance $\sigma^2 = 0.01$. In the left panel, the linear GP using the MCMC approach ESS (linGP-ESS for short) and developed in Sect. 3.1 is applied for generating 6,000 MCMC sample paths, where the first 1,000 are discarded as burn-in. The gray shaded area represents the 95% confidence interval obtained from the 6,000 sample paths. The blue dashed curve represents the posterior mean using Equation (9). The mean square error (MSE) as well as the running time in seconds for generating the posterior sample paths are displayed in the main of the panels. In the right panel, the proposed approach using MUR for large datasets (linGP-bData-MUR for short) developed in Sect. 3.2 is employed. It is worth noting that both approaches fit well the data (green stars) with a comparable MSE. However, the linGP-bData-MUR outperforms the linGP-ESS in terms of computational running time. This is due to the fact that the linGP-bData-MUR eliminates the necessity of evaluating a likelihood function at each MCMC iteration. Now, we evaluate their performance for a large number of observations.

In Fig. 3, the runtime in seconds for generating 15,000 sample paths using the two developed methods is shown. The computational running time of the two approaches grows linearly as a function of the number of samples, but with different slopes. As expected, the proposed linGP-bData-MUR method outperforms the linGP-ESS approach for large values of sample size n. This is because linGP-bData-MUR avoids evaluating the likelihood function at each MCMC iteration. Furthermore, unlike the linGP-ESS approach, the linGP-bData-MUR method provides an exact solution for generating the posterior distribution (see Fig. 4).

In Fig. 4, we applied the two methods to an extreme case, where the samples are concentrated in the first half of the input domain $\mathcal{X}$ and only a few observations ($n = 10$) are available in the second half. The stationary Matérn covariance function (2) with a smoothness parameter $\nu = 5/2$ is employed. The prior $\boldsymbol{\xi}$ is generated using Cholesky factorization [24, Sect. 3.3] even though the precision covariance matrix $\boldsymbol{K}$ is Toeplitz. This is because the dimension N is reasonably 'small', and thus, Cholesky factorization

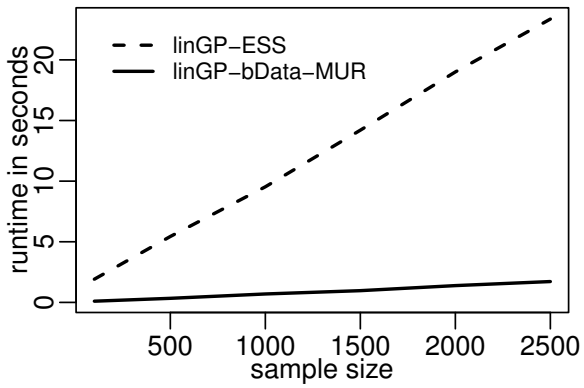

Fig. 3. Runtime in seconds for generating 15,000 posterior sample paths as a function of the number of samples for the two competing approaches

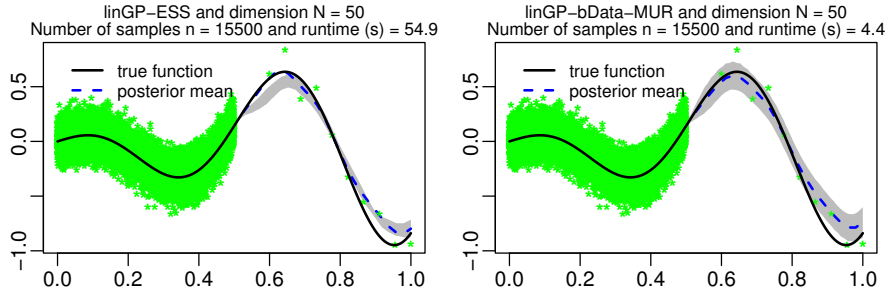

Fig. 4. Same settings as Fig. 2 except the number of samples n which is fixed at 15,500 instead of 100. Unlike linGP-bDatMUR, the 95% confidence interval (gray shaded area) of the linGP-ESS approach does not closely follow the posterior mean (blue dashed curve)

is highly efficient. For both approaches, the posterior mean (blue dashed curve) defined in Equation (9) fits the data well. However, unlike the linGP-bData-MUR method, the 95% confidence interval (gray shaded area) of the linGP-ESS approach, which is computed from 6,000 simulations, fails to follow the posterior mean in the second part of the domain. It is worth noting that this 95% confidence interval is significantly reduced for both approaches in the first part of the domain (i.e., when $x \leq 0.5$) due to the high number of observations. Finally, we note that when the $n = 15,500$ data (green stars) uniformly cover the entire domain $\mathcal{X}$, the issue of convergence in the MCMC approach is no longer present.

5 Real application

In this section, we apply the two strategies developed in this paper to real-world diamond data. This dataset consists of the prices in US dollars (326$-18,823$) of $n = 53,940$ diamonds as a function of their carat, i.e., weight of the diamond (0.2-5.01). This application is particularly interesting because the dataset is divided into two parts of

the domain: one with a large number of observations in the first part and another with relatively few observations in the latter part (green stars in Fig. 6). Before using the entire dataset, we first test the two developed approaches on a smaller subset of training samples. The stationary Matérn covariance function (2) is employed with a smoothness parameter $\nu = 5/2$.

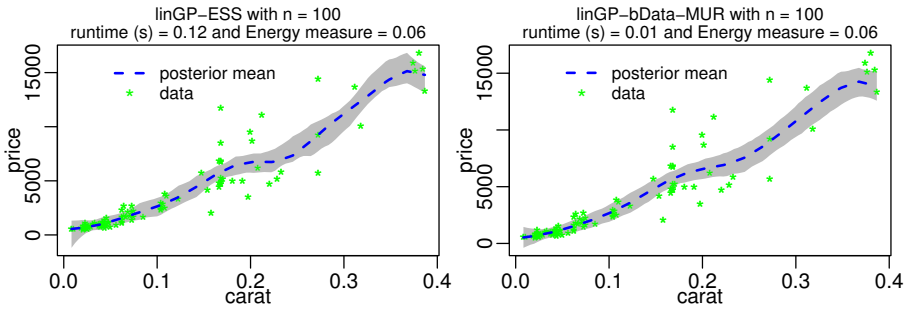

Fig. 5. Accuracy estimation of the price of only $n = 100$ diamonds as a function of carat. The two developed strategies are employed linGP-ESS (left panel) and linGP-bData-MUR (right panel). The gray shaded area represents the 95% confidence interval based on 1,000 sample paths. The computational running time of generating 1,000 sample paths and the energy measure criterion are displayed in the main of each panel

In Fig. 5, the two strategies developed in this paper are employed to estimate the prices of $n = 100$ diamonds as a function of their carat (standardized to fall in the range $\mathcal{X} = [0, 1]$). We randomly choose only $n = 100$ diamonds to test the two approaches for a few observations (green stars). The blue dashed curve represents the posterior mean (9), while the gray shaded area represents the 95% confidence interval based on 1,000 posterior sample paths. The Energy measure criterion is computed as follows: $\sum_{i=1}^{100}(y_i - \hat{y}_i)^2 / \sum_{i=1}^{100} y_i^2$, where $\hat{y}_i$ represents the posterior mean (9) evaluated at the training covariate x_i. It is worth noting that the linGP-bData-MUR approach outperforms the linGP-ESS method in terms of the computational time required to generate 1,000 sample paths.

Now, in Fig. 6, we evaluate the performance of these two approaches in terms of both computational running time and prediction accuracy when using the entire dataset. The green stars represent the $n = 53,940$ samples, which are concentrated in the first half of the input domain $\mathcal{X}$ (i.e., the diamond's weight between 0.2 and 3). This setup provides an interesting scenario to evaluate the convergence of the two approaches. It is evident that the proposed linGP-bData-MUR method is significantly faster than the MCMC-based linGP-ESS approach in generating 1,000 posterior sample paths. Additionally, as expected, in the second half of the domain, the posterior distribution (gray shaded area) produced by linGP-bData-MUR fits the data well, unlike the linGP-ESS method. This difference arises because linGP-bData-MUR is an exact method, whereas linGP-ESS is an approximate method that fails to adequately capture the data in the second half of the domain $\mathcal{X}$.

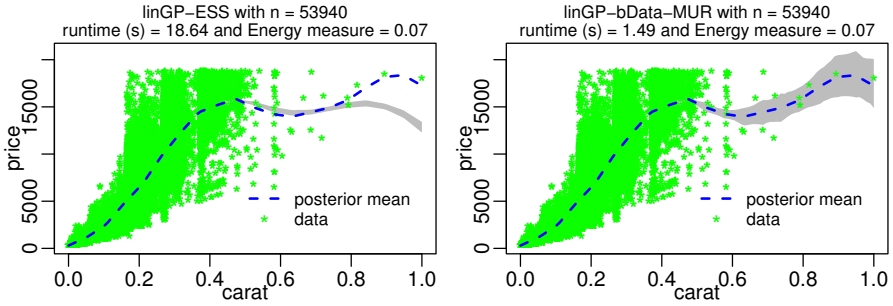

Fig. 6. Accuracy estimation of the price of $n = 53,940$ diamonds as a function of carat. The two developed strategies are employed linGP-ESS (left panel) and linGP-bData-MUR (right panel). The computational running time of generating 1,000 sample paths and the energy measure criterion are displayed in the main of each panel

6 Conclusion

In this paper, Bayesian nonparametric function estimation for a large number of observations is studied. Two distinct strategies for generating finite-dimensional Bayesian linear models are developed. The first approach employs MCMC with Elliptical Slice Sampling (ESS) and is considered an approximate method. The second approach, which is exact, modifies Matheron's update rule (MUR) using Bayes' rule. We theoretically demonstrate that MUR can be adapted for large datasets, significantly reducing computational complexity. The excellent performance of these two methods, particularly in terms of computational runtime and prediction accuracy, is evaluated in the context of Bayesian nonparametric function estimation using both synthetic and real datasets. Based on our numerical experiments, the proposed MUR approach is more stable and faster than the approach based on the MCMC with ESS.

References

1. S. Banerjee, B.P. Carlin, and A.E. Gelfand. *Hierarchical modeling and analysis for spatial data*. Chapman and Hall/CRC, 2003.
2. P.M. Chak, N. Madras, and B. Smith. Semi-nonparametric estimation with Bernstein polynomials. *Econ. Lett.*, 89(2):153–156, 2005.
3. I.-S. Chang, C.A. Hsiung, Y.-J. Wu, and C.-C. Yang. Bayesian survival analysis using Bernstein polynomials. *Scand. J. Stat.*, 32(3):447–466, 2005.
4. Y. Cong, B. Chen, and M. Zhou. Fast simulation of hyperplane-truncated multivariate normal distributions. *Bayesian Anal.*, 12(4):1017–1037, 2017.
5. S.L. Cotter, G.O. Roberts, A.M. Stuart, and D. White. MCMC methods for functions: modifying old algorithms to make them faster. *Stat. Sci.*, 28(3):424–446, 2013.
6. S.M. Curtis and S.K. Ghosh. A variable selection approach to monotonic regression with Bernstein polynomials. *J. Appl. Stat.*, 38(5):961–976, 2011.
7. C. De Boor. Package for calculating with B-splines. *SIAM J. Numer. Anal.*, 14(3):441–472, 1977.

8. N. Durrande, V. Adam, L. Bordeaux, S. Eleftheriadis, and J. Hensman. Banded matrix operators for Gaussian Markov models in the automatic differentiation era. In *The 22nd International Conference on AISTATS*, volume 89, pages 2780–2789. PMLR, 2019.
9. P.H.C. Eilers and B.D. Marx. Flexible smoothing with B-splines and penalties. *Stat. Sci.*, 11(2):89–121, 1996.
10. M.G. Genton. Classes of kernels for machine learning: a statistics perspective. *JMLR*, 2:299–312, 2001.
11. G. Golub and C.F. Van Loan. *Matrix computations*. The Johns Hopkins University Press, 1996.
12. A.G. Journel. *Simulations conditionnelles de gisements miniers*. PhD thesis, Université de Nancy-I, Nancy France, 1974.
13. A.G. Journel and C.J. Huijbregts. *Mining geostatistics*. Academic Press, 1976.
14. M. Loève. Elementary probability theory. In *Probability Theory I*, pages 1–52. Springer, 1977.
15. H. Maatouk and X. Bay. Gaussian process emulators for computer experiments with inequality constraints. *Math. Geosci.*, 49(5):557–582, 2017.
16. H. Maatouk, D. Rullière, and X. Bay. Bayesian analysis of constrained Gaussian processes. *Bayesian Anal.*, pages 1–30, 2024.
17. H. Maatouk, D. Rullière, and X. Bay. Efficient constrained Gaussian process approximation using elliptical slice sampling. working paper or preprint, March 2024.
18. H. Maatouk, D. Rullière, and X. Bay. Large scale Gaussian processes with Matheron's update rule and Karhunen-Loève expansion. In Aicke Hinrichs, Peter Kritzer, and Friedrich Pillichshammer, editors, *Monte Carlo and Quasi-Monte Carlo Methods*, pages 469–487, Cham, 2024. Springer International Publishing.
19. H. Maatouk, D. Rullière, and X. Bay. Sampling large hyperplane-truncated multivariate normal distributions. *Comput. Stat.*, 39(4):1779–1806, 2024.
20. H. Maatouk, D. Rullière, and X. Bay. Large-scale constrained Gaussian processes for shape-restricted function estimation. *Stat. Comput.*, 35(7), 2025.
21. I. Murray, R. Adams, and D. MacKay. Elliptical slice sampling. In *Proceedings of the thirteenth international conference on AISTATS*, pages 541–548. JMLR Workshop and Conference Proceedings, 2010.
22. R.M. Neal. Regression and classification using Gaussian process priors. *Bayesian Statistics*, 6:475–501, 1999.
23. J. Sacks, W.J. Welch, T.J. Mitchell, and H.P. Wynn. Design and analysis of computer experiments. *Statist. Sci.*, 4(4):409–423, 1989.
24. R.A. Thisted. *Elements of Statistical Computing*. New York: Chapman and Hall, 1988.
25. L. Wang. *Karhunen-Loève expansions and their applications*. London School of Economics and Political Science (United Kingdom), 2008.
26. C.K. Williams and C.E. Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.
27. J.T. Wilson, V. Borovitskiy, A. Terenin, P. Mostowsky, and M.P. Deisenroth. Pathwise conditioning of Gaussian processes. *JMLR*, 22(105):1–47, 2021.
28. A.T.A. Wood and G. Chan. Simulation of stationary Gaussian processes in $[0,1]^d$. *J. Comput. Graph. Stat.*, 3(4):409–432, 1994.
29. D.L. Zimmerman. Computationally exploitable structure of covariance matrices and generalized convariance matrices in spatial models. *J. Stat. Comput. Simul.*, 32(1-2):1–15, 1989.