Noname manuscript No.
(will be inserted by the editor)

Practical Hilbert space approximate Bayesian Gaussian processes for probabilistic programming

Gabriel Riutort-Mayol · Paul-Christian Bürkner · Michael R. Andersen · Arno Solin · Aki Vehtari

Received: date / Accepted: date

Abstract Gaussian processes are powerful non-parametric probabilistic models for stochastic functions. However, they entail a complexity that is computationally intractable when the number of observations is large, especially when estimated with fully Bayesian methods such as Markov chain Monte Carlo. In this paper, we focus on a novel approach for low-rank approximate Bayesian Gaussian processes, based on a basis function approximation via Laplace eigenfunctions for stationary covariance functions. The main contribution of this paper is a detailed analysis of the performance and practical implementation of the method in relation to key factors such as the number of basis functions, domain of the prediction space, and smoothness of the latent function. We provide intuitive visualizations and recommendations for choosing the values of these factors, which make it easier for users to improve approximation accuracy and computational performance. We also propose diagnostics for checking that the number of basis functions and the domain of the prediction space are adequate given the data. The proposed approach is simple and exhibits an attractive computational complexity due to its linear structure, and it is easy to implement in probabilistic programming frameworks. Several illustrative examples of the performance and applicability of the method in the probabilistic programming language Stan are presented together with the underlying Stan model code.

Keywords Gaussian process · Low-rank Gaussian process · Hilbert space methods · Sparse Gaussian process · Bayesian statistics · Stan

Gabriel Riutort-Mayol
Department of Cartographic Engineering, Geodesy, and Photogrammetry, Universitat Politècnica de València, Spain
E-mail: gabriuma@gmail.com

Paul-Christian Bürkner
Excellence Cluster for Simulation Technology, University of Stuttgart, Germany
Most of work was done while at Dept. of Computer Science at Aalto University

Michael R. Andersen
Department of Applied Mathematics and Computer Science, Technical University of Denmark, Denmark

Arno Solin
Department of Computer Science, Aalto University, Finland

Aki Vehtari
Department of Computer Science, Aalto University, Finland

1 Introduction

Gaussian processes (GPs) are flexible statistical models for specifying probability distributions over multi-dimensional non-linear functions (Rasmussen and Williams, 2006; Neal, 1997). Their name stems from the fact that any finite set of function values is jointly distributed as a multivariate Gaussian distribution. GPs are defined by a mean and a covariance function. The covariance function encodes our prior assumptions about the functional relationship, such as continuity, smoothness, periodicity and scale properties. GPs not only allow for non-linear effects but can also implicitly handle interactions between input variables (covariates). Different types of covariance functions can be combined for further increased flexibility. Due to their generality and flexibility, GPs are of broad interest across machine learning and statistics (Rasmussen and Williams, 2006; Neal, 1997). Among others, they find application in the fields of spatial epidemiology (Diggle, 2013; Carlin et al., 2014), robotics and control (Deisenroth et al., 2015), signal processing (Särkkä et al., 2013), neuroimaging (Andersen et al., 2017) as well as Bayesian optimization and probabilistic numerics (Roberts, 2010; Briol et al., 2015; Hennig et al., 2015).

The key element of a GP is the covariance function that defines the dependence structure between function values at

different inputs. However, computing the posterior distribution of a GP comes with a computational issue because of the need to invert the covariance matrix. Given n observations in the data, the computational complexity and memory requirements of computing the posterior distribution for a GP in general scale as $O(n^3)$ and $O(n^2)$, respectively. This limits their application to rather small data sets of a few tens of thousands observations at most. The problem becomes more severe when performing full Bayesian inference via sampling methods, where in each sampling step we need $O(n^3)$ computations when inverting the Gram matrix of the covariance function, usually through Cholesky factorization. To alleviate these computational demands, several approximate methods have been proposed.

Sparse GPs are based on low-rank approximations of the covariance matrix. The low-rank approximation with $m \ll n$ *inducing points* implies reduced memory requirements of $O(nm)$ and corresponding computational complexity of $O(nm^2)$. A unifying view on sparse GPs based on approximate generative methods is provided by Quiñonero-Candela and Rasmussen (2005), while a general review is provided by Rasmussen and Williams (2006). Burt et al. (2019) show that for regression with normally distributed covariates in D dimensions and using the squared exponential covariance function, $M = O(\log^D N)$ is sufficient for an accurate approximation. An alternative class of low-rank approximations is based on forming a basis function approximation with $m \ll n$ basis functions. The basis functions are usually presented explicitly, but can also be used to form a low-rank covariance matrix approximation. Common basis function approximations rest on the spectral analysis and series expansions of GPs (Loève, 1977; Van Trees, 1968; Adler, 1981; Cramér and Leadbetter, 2013). Sparse spectrum GPs are based on a sparse approximation to the frequency domain representation of a GP (Lázaro Gredilla, 2010; Quiñonero-Candela et al., 2010; Gal and Turner, 2015). Recently, Hensman et al. (2017) presented a variational Fourier feature approximation for GPs that was derived for the Matérn class of kernels. Another related method for approximating kernels relies on random Fourier features (Rahimi and Recht, 2008, 2009). Certain spline smoothing basis functions are equivalent to GPs with certain covariance functions (Wahba, 1990; Furrer and Nychka, 2007). Recent related work based on a spectral representation of GPs as an infinite series expansion with the Karhunen-Loève representation (see, e.g., Grenander, 1981) is presented by Jo et al. (2019).

Aki: we could cite in the intro https://arxiv.org/abs/2111.01084 as a paper which reviews alternative methods

In this paper, we focus on a recent framework for fast and accurate inference for fully Bayesian GPs using basis function approximations based on approximation via Laplace eigenfunctions for stationary covariance functions proposed by Solin and Särkkä (2020). Using a basis function expansion, a GP is approximated with a linear model which makes inference considerably faster. The linear model structure makes GPs easy to implement as building blocks in more complicated models in modular probabilistic programming frameworks, where there is a big benefit if the approximation specific computation is simple. Furthermore, a linear representation of a GP makes it easier to be used as latent function in non-Gaussian observational models allowing for more modelling flexibility. The basis function approximation via Laplace eigenfunctions can be made arbitrary accurate and the trade-off between computational complexity and approximation accuracy can easily be controlled.

The Laplace eigenfunctions can be computed analytically and they are independent of the particular choice of the covariance function including the hyperparameters. While the pre-computation cost of the basis functions is $O(m^2 n)$, the computational cost of learning the covariance function parameters is $O(mn + m)$ in every step of the optimizer or sampler. This is a big advantage in terms of speed for iterative algorithms such as Markov chain Monte Carlo (MCMC). Another advantage is the reduced memory requirements of automatic differentiation methods used in modern probabilistic programming frameworks, such as Stan (Carpenter et al., 2017) and others. This is because the memory requirements of automatic differentiation scale with the size of the autodiff expression tree which in direct implementations is simpler for basis function than covariance matrix based approach. The basis function approach also provides an easy way to apply a non-centered parameterization of GPs, which reduces the posterior dependency between parameters representing the estimated function and the hyperparameters of the covariance function, which further improves MCMC efficiency.

While Solin and Särkkä (2020) have fully developed the mathematical theory behind this specific approximation of GPs, further work is needed for its practical implementation in probabilistic programming frameworks. In this paper, the interactions among the key factors of the method such as the number of basis functions, domain of the prediction space, and properties of the true functional relationship between covariates and response variable, are investigated and analyzed in detail in relation to the computational performance and accuracy of the method. Practical recommendations are given for the values of the key factors based on intuitive graphical summaries that encode the recognized relationships. Our recommendations will help users to choose valid and optimized values for these factors, improving computational performance without sacrificing modeling accuracy. We also propose diagnostics to indicate whether the chosen values for the number of basis functions and the domain of the prediction space are adequate to model the data well.

We have implemented the approach in the probabilistic programming language Stan (Carpenter et al., 2017) as well as subsequently in the *brms* package (Bürkner, 2017) of the R software (R Core Team, 2019). Several illustrative examples of the performance and applicability of the method are shown using both simulated and real datasets. All examples are accompanied by the corresponding Stan code. Although there are several GP specific software packages available to date, for example, GPML (Rasmussen and Nickisch, 2010), GPstuff (Vanhatalo et al., 2013), GPy (GPy, 2012), and GPflow (Matthews et al., 2017), each provide efficient implementations only for a restricted range of GP-based models. In this paper, we do not focus on the fastest possible inference for a small set of specific GP models, but instead we are interested in how GPs can be easily used as modular components in probabilistic programming frameworks.

The remainder of the paper is structured as follows. In Section 2, we introduce GPs, covariance functions and spectral density functions. In Section 3, the reduced-rank approximation to GPs proposed by Solin and Särkkä (2020) is described. In Section 4, the accuracy of these approximations under several conditions using analytical and numerical methods is analyzed. Several case studies in which we fit exact and approximate GPs to real and simulated data are provided in Section 5. A brief conclusion of the work is made in Section 6. Appendix A includes a brief presentation of the mathematical details behind the Hilbert space approximation of a stationary covariance function, and Appendix B presents a low-rank representation of a GP for the particular case of a periodic covariance function. Online supplemental material with more case studies illustrating the performance and applicability of the method can be found online at `https://github.com/gabriuma/basis_functions_approach_to_GP` in the subfolder `Paper/online_supplemental_material`.

2 Gaussian process as a prior

A GP is a stochastic process which defines the distribution of a collection of random variables indexed by a continuous variable, that is, $\{f(t) : t \in \mathcal{T}\}$ for some index set $\mathcal{T}$. GPs have the defining property that the marginal distribution of any finite subset of random variables, $\{f(t_1), f(t_2), \ldots, f(t_N)\}$, is a multivariate Gaussian distribution.

In this work, GPs will take the role of a prior distribution over function spaces for non-parametric latent functions in a Bayesian setting. Consider a data set $\mathcal{D} = \{(\boldsymbol{x}_n, y_n)\}_{n=1}^N$, where y_n is modelled conditionally as $p(y_n \mid f(\boldsymbol{x}_n), \phi)$, where p is some parametric distribution with parameters ϕ, and f is an unknown function with a GP prior, which depends on an input $\boldsymbol{x}_n \in \mathbb{R}^D$. This generalizes readily to more complex models depending on several unknown functions, for example such as $p(y_n \mid f(\boldsymbol{x}_n), g(\boldsymbol{x}_n))$ or multilevel models. Our goal is to obtain the posterior distribution for the value of the function $\tilde{f} = f(\tilde{\boldsymbol{x}})$ evaluated at a new input point $\tilde{\boldsymbol{x}}$.

We assume a GP prior for $f \sim \mathcal{GP}(\mu(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}'))$, where $\mu : \mathbb{R}^D \to \mathbb{R}$ and $k : \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$ are the mean and covariance functions, respectively,

$$\mu(\boldsymbol{x}) = \mathbb{E}[f(\boldsymbol{x})]\,,$$
$$k(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}[(f(\boldsymbol{x}) - \mu(\boldsymbol{x}))\,(f(\boldsymbol{x}') - \mu(\boldsymbol{x}'))]\,.$$

The mean and covariance functions completely characterize the GP prior, and control the a priori behavior of the function f. Let $\boldsymbol{f} = \{f(\boldsymbol{x}_n)\}_{n=1}^N$, then the resulting prior distribution for $\boldsymbol{f}$ is a multivariate Gaussian distribution $\boldsymbol{f} \sim \mathrm{Normal}(\boldsymbol{\mu}, \boldsymbol{K})$, where $\boldsymbol{\mu} = \{\mu(\boldsymbol{x}_n)\}_{n=1}^N$ is the mean and $\boldsymbol{K}$ the covariance matrix, where $K_{i,j} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$. In the following, we focus on zero-mean Gaussian processes, that is set $\mu(\boldsymbol{x}) = 0$. The covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$ might depend on a set of hyperparameters, $\boldsymbol{\theta}$, but we will not write this dependency explicitly to ease the notation. The joint distribution of $\boldsymbol{f}$ and a new $\tilde{f}$ is also a multivariate Gaussian as,

$$p(\boldsymbol{f}, \tilde{f}) = \mathrm{Normal}\left(\begin{bmatrix} \boldsymbol{f} \\ f^* \end{bmatrix} \middle| \, \boldsymbol{0}, \begin{bmatrix} \boldsymbol{K}_{\boldsymbol{f},\boldsymbol{f}} & \boldsymbol{k}_{\boldsymbol{f},\tilde{f}} \\ \boldsymbol{k}_{\tilde{f},\boldsymbol{f}} & k_{\tilde{f},\tilde{f}} \end{bmatrix}\right),$$

where $\boldsymbol{k}_{\boldsymbol{f},\tilde{f}}$ is the covariance between $\boldsymbol{f}$ and $\tilde{f}$, and $k_{\tilde{f},\tilde{f}}$ is the prior variance of $\tilde{f}$.

If $p(y_n \mid f(\boldsymbol{x}_n), \phi) = \mathrm{Normal}(y_n \mid f(\boldsymbol{x}_n), \sigma)$ then $\boldsymbol{f}$ can be integrated out analytically (with a computational cost of $O(n^3)$ for exact GPs and $O(nm^2)$ for sparse GPs). If $p(y_n \mid f(\boldsymbol{x}_n), g(\boldsymbol{x}_n)) = \mathrm{Normal}(y_n \mid f(\boldsymbol{x}_n), g(\boldsymbol{x}_n))$ or $p(y_n \mid f(\boldsymbol{x}_n), \phi)$ is non-Gaussian, the marginalization does not have a closed-form solution. Furthermore, if a prior distribution is imposed on ϕ and $\boldsymbol{\theta}$ to form a joint posterior for ϕ, $\boldsymbol{\theta}$ and $\boldsymbol{f}$, approximate inference such as Markov chain Monte Carlo (MCMC; Brooks et al., 2011), Laplace approximation (Williams and Barber, 1998; Rasmussen and Williams, 2006), expectation propagation (Minka, 2001), or variational inference methods (Gibbs and MacKay, 2000; Csató et al., 2000) are required. In this paper, we focus on the use of MCMC for integrating over the joint posterior. MCMC is usually not the fastest approach, but it is flexible and allows accurate inference and uncertainty estimates for general models in probabilistic programming settings. We consider the computational costs of GPs specifically from this point of view.

2.1 Covariance functions and spectral density

The covariance function is the crucial ingredient in a GP as it encodes our prior assumptions about the function, and characterizes the correlations between function values at different

locations in the input space. A covariance function needs to be symmetric and positive semi-definite (Rasmussen and Williams, 2006). A stationary covariance function is a function of $\boldsymbol{\tau} = \boldsymbol{x} - \boldsymbol{x}' \in \mathbb{R}^D$, such that it can be written $k(\boldsymbol{x}, \boldsymbol{x}') = k(\boldsymbol{\tau})$, which means that the covariance is invariant to translations. Isotropic covariance functions depend only on the input points through the norm of the difference, $k(\boldsymbol{x}, \boldsymbol{x}') = k(|\boldsymbol{x} - \boldsymbol{x}'|) = k(r), r \in \mathbb{R}$, which means that the covariance is both translation and rotation invariant. The most commonly used distance between observations is the L2-norm ($|\boldsymbol{x} - \boldsymbol{x}'|_{L2}$), also known as Euclidean distance, although other types of distances can be considered.

The Matérn class of isotropic covariance functions is given by,

$$k_\nu(r) = \alpha \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu} r}{\ell} \right)^\nu K_\nu \left(\frac{\sqrt{2\nu} r}{\ell} \right),$$

where $\nu > 0$ is the order the kernel, K_ν the modified Bessel function of the second kind, and the $\ell > 0$ and $\alpha > 0$ are the length-scale and magnitude (marginal variance), respectively, of the kernel. The particular case where $\nu = \infty$, $\nu = 1/2$, $\nu = 3/2$ and $\nu = 5/2$ are probably the most commonly used kernels (Rasmussen and Williams, 2006),

$$\begin{aligned}
k_\infty(r) &= \alpha \exp\left(-\frac{1}{2} \frac{r^2}{\ell^2} \right), \\
k_{\frac{1}{2}}(r) &= \alpha \exp\left(-\frac{r}{\ell} \right), \\
k_{\frac{3}{2}}(r) &= \alpha \left(1 + \frac{\sqrt{3} r}{\ell} \right) \exp\left(-\frac{\sqrt{3} r}{\ell} \right), \\
k_{\frac{5}{2}}(r) &= \alpha \left(1 + \frac{\sqrt{5} r}{\ell} + \frac{5 r^2}{3\ell^2} \right) \exp\left(-\frac{\sqrt{5} r}{\ell} \right).
\end{aligned}$$

The former is commonly known as the squared exponential or exponentiated quadratic covariance function. Assuming the Euclidean distance between observations, $r = |\boldsymbol{x} - \boldsymbol{x}'|_{L2} = \sqrt{\sum_{i=1}^D (x_i - x_i')^2}$, the kernels written above take the form

$$\begin{aligned}
k_\infty(r) &= \alpha \exp\left(-\frac{1}{2} \sum_{i=1}^D \frac{(x_i - x_i')^2}{\ell_i^2} \right), \\
k_{\frac{1}{2}}(r) &= \alpha \exp\left(-\sum_{i=1}^D \frac{(x_i - x_i')}{\ell_i} \right), \\
k_{\frac{3}{2}}(r) &= \alpha \left(1 + \sqrt{\sum_{i=1}^D \frac{3(x_i - x_i')^2}{\ell_i^2}} \right) \\
&\quad \times \exp\left(-\sqrt{\sum_{i=1}^D \frac{3(x_i - x_i')^2}{\ell_i^2}} \right), \\
k_{\frac{5}{2}}(r) &= \alpha \left(1 + \sqrt{\sum_{i=1}^D \frac{3(x_i - x_i')^2}{\ell_i^2}} + \frac{5}{3} \sum_{i=1}^D \frac{(x_i - x_i')^2}{\ell_i^2} \right) \\
&\quad \times \exp\left(-\sqrt{\sum_{i=1}^D \frac{5(x_i - x_i')^2}{\ell_i^2}} \right).
\end{aligned}$$

Notice that the previous expressions, $k_\infty(r)$, $k_{\frac{1}{2}}(r)$, $k_{\frac{3}{2}}(r)$ and $k_{\frac{5}{2}}(r)$, have been easily generalized to using a multidimensional length-scale $\boldsymbol{\ell} \in \mathbb{R}^D$. Using individual length-scales for each dimension turns the isotropic covariance function into a non-isotropic covariance function. That is, for a non-isotropic covariance function, the smoothness may vary across different input dimensions.

Stationary covariance functions can be represented in terms of their spectral densities (see, e.g., Rasmussen and Williams, 2006). In this sense, the covariance function of a stationary process can be represented as the Fourier transform of a positive finite measure (*Bochner's theorem*; see, e.g., Akhiezer and Glazman, 1993). If this measure has a density, it is known as the spectral density of the covariance function, and the covariance function and the spectral density are Fourier duals, known as the *Wiener-Khintchine theorem* (Rasmussen and Williams, 2006). The spectral density functions associated with the Matérn class of covariance functions are given by

$$S_\nu(\boldsymbol{\omega}) = \alpha \frac{2^D \pi^{D/2} \Gamma(\nu + D/2)(2\nu)^\nu}{\Gamma(\nu)\, \ell^{2\nu}} \left(\frac{2\nu}{\ell^2} + 4\pi^2 \boldsymbol{\omega}^\intercal \boldsymbol{\omega} \right)^{-(\nu + D/2)}$$

in D dimensions, where vector $\boldsymbol{\omega} \in \mathbb{R}^D$ is in the frequency domain, and ℓ and α are the length-scale and magnitude (marginal variance), respectively, of the kernel. The particular cases, where $\nu = \infty$, $\nu = 1/2$, $\nu = 3/2$ and $\nu = 5/2$, take

the form

$$S_\infty(\boldsymbol{\omega}) = \alpha\,(\sqrt{2\pi})^D \ell^D \exp(-\frac{1}{2}\ell^2\boldsymbol{\omega}^\intercal\boldsymbol{\omega}), \tag{1}$$

$$S_{\frac{1}{2}}(\boldsymbol{\omega}) = \alpha\,\frac{2^D\pi^{D/2}\Gamma(\frac{D+1}{2})}{\sqrt{\pi}\ell}\left(\frac{1}{\ell^2}+\boldsymbol{\omega}^\intercal\boldsymbol{\omega}\right)^{-\frac{D+1}{2}}, \tag{2}$$

$$S_{\frac{3}{2}}(\boldsymbol{\omega}) = \alpha\,\frac{2^D\pi^{D/2}\Gamma(\frac{D+3}{2})3^{3/2}}{\frac{1}{2}\sqrt{\pi}\ell^3}\left(\frac{3}{\ell^2}+\boldsymbol{\omega}^\intercal\boldsymbol{\omega}\right)^{-\frac{D+3}{2}}, \tag{3}$$

$$S_{\frac{5}{2}}(\boldsymbol{\omega}) = \alpha\,\frac{2^D\pi^{D/2}\Gamma(\frac{D+5}{2})5^{5/2}}{\frac{3}{4}\sqrt{\pi}\ell^5}\left(\frac{5}{\ell^2}+\boldsymbol{\omega}^\intercal\boldsymbol{\omega}\right)^{-\frac{D+5}{2}}. \tag{4}$$

For instance, with input dimensionality $D = 3$ and $\boldsymbol{\omega} = (\omega_1, \omega_2, \omega_3)^\intercal$, the spectral densities written above take the form

$$S_\infty(\boldsymbol{\omega}) = \alpha\,(2\pi)^{3/2}\prod_{i=1}^{3}\ell_i \exp\left(-\frac{1}{2}\sum_{i=1}^{3}\ell_i^2\omega_i^2\right),$$

$$S_{\frac{1}{2}}(\boldsymbol{\omega}) = \alpha\,8\,\pi\prod_{i=1}^{3}\ell_i\left(1+\sum_{i=1}^{3}\ell_i^2\omega_i^2\right)^{-2},$$

$$S_{\frac{3}{2}}(\boldsymbol{\omega}) = \alpha\,32\,\pi 3^{3/2}\prod_{i=1}^{3}\ell_i\left(3+\sum_{i=1}^{3}\ell_i^2\omega_i^2\right)^{-3},$$

$$S_{\frac{5}{2}}(\boldsymbol{\omega}) = \alpha\,\frac{64}{3}\,\pi 5^{5/2}\prod_{i=1}^{3}\ell_i\left(5+\sum_{i=1}^{3}\ell_i^2\omega_i^2\right)^{-4}.$$

where individual length-scales ℓ_i for each frequency dimension ω_i have been used.

3 Hilbert space approximate Gaussian process model

The approximate GP method, developed by Solin and Särkkä (2020) and further analysed in this paper, is based on considering the covariance operator of a stationary covariance function as a pseudo-differential operator constructed as a series of Laplace operators. Then, the pseudo-differential operator is approximated with *Hilbert space* methods on a compact subset $\Omega \subset \mathbb{R}^D$ subject to boundary conditions. For brevity, we will refer to these approximate Gaussian processes as HSGPs. Below, we will present the main results around HSGPs relevant for practical applications. More details on the theoretical background are provided by Solin and Särkkä (2020). Our starting point for presenting the method is the definition of the covariance function as a series expansion of eigenvalues and eigenfunctions of the Laplacian operator. The mathematical details of this approximation are briefly presented in Appendix A.

3.1 Unidimensional GPs

We begin by focusing on the case of a unidimensional input space (i.e., on GPs with just a single covariate) such that $\Omega \in [-L, L] \subset \mathbb{R}$, where L is some positive real number to which we also refer as boundary condition. As Ω describes the interval in which the approximations are valid, L plays a critical role in the accuracy of HSGPs. We will come back to this issue in Section 4.

Within Ω, we can write any stationary covariance function with input values $x, x' \in \Omega$ as

$$k(x, x') = \sum_{j=1}^{\infty} S_\theta(\sqrt{\lambda_j})\phi_j(x)\phi_j(x'), \tag{5}$$

where S_θ is the spectral density of the stationary covariance function k (see Section 2.1) and θ is the set of hyperparameters of k (Rasmussen and Williams, 2006). The terms $\{\lambda_j\}_{j=1}^{\infty}$ and $\{\phi_j(x)\}_{j=1}^{\infty}$ are the sets of eigenvalues and eigenvectors, respectively, of the Laplacian operator in the given domain Ω. Namely, they satisfy the following eigenvalue problem in Ω when applying the Dirichlet boundary condition (other boundary conditions could be used as well)

$$\begin{aligned} -\nabla^2\phi_j(x) &= \lambda_j\phi_j(x), && x \in \Omega \\ \phi_j(x) &= 0, && x \notin \Omega. \end{aligned} \tag{6}$$

The eigenvalues $\lambda_j > 0$ are real and positive because the Laplacian is a positive definite Hermitian operator, and the eigenfunctions ϕ_j for the eigenvalue problem in eq. (6) are sinusoidal functions. The solution to the eigenvalue problem is independent of the specific choice of covariance function and is given by

$$\lambda_j = \left(\frac{j\pi}{2L}\right)^2, \tag{7}$$

$$\phi_j(x) = \sqrt{\frac{1}{L}}\,\sin\Big(\sqrt{\lambda_j}(x+L)\Big). \tag{8}$$

If we truncate the sum in eq. (5) to the first m terms, the approximate covariance function becomes

$$k(x, x') \approx \sum_{j=1}^{m} S_\theta(\sqrt{\lambda_j})\phi_j(x)\phi_j(x') = \boldsymbol{\phi}(x)^\intercal\boldsymbol{\Delta}\boldsymbol{\phi}(x'),$$

where $\boldsymbol{\phi}(x) = \{\phi_j(x)\}_{j=1}^{m} \in \mathbb{R}^m$ is the column vector of basis functions, and $\boldsymbol{\Delta} \in \mathbb{R}^{m\times m}$ is a diagonal matrix of the spectral density evaluated at the square root of the eigenvalues, that is, $S_\theta(\sqrt{\lambda_j})$,

$$\boldsymbol{\Delta} = \begin{bmatrix} S_\theta(\sqrt{\lambda_1}) & & \\ & \ddots & \\ & & S_\theta(\sqrt{\lambda_m}) \end{bmatrix}.$$

Thus, the Gram matrix $\boldsymbol{K}$ for the covariance function k for a set of observations $i = 1, \dots, n$ and corresponding input values $\{x_i\}_{i=1}^n$ can be represented as

$$\boldsymbol{K} = \boldsymbol{\Phi}\boldsymbol{\Delta}\boldsymbol{\Phi}^\intercal,$$

where $\boldsymbol{\Phi} \in \mathbb{R}^{n \times m}$ is the matrix of eigenfunctions $\phi_j(x_i)$

$$\boldsymbol{\Phi} = \begin{bmatrix} \phi_1(x_1) & \cdots & \phi_m(x_1) \\ \vdots & \ddots & \vdots \\ \phi_1(x_n) & \cdots & \phi_m(x_n) \end{bmatrix}.$$

As a result, the model for f can be written as

$$\boldsymbol{f} \sim \text{Normal}(\boldsymbol{\mu}, \boldsymbol{\Phi}\boldsymbol{\Delta}\boldsymbol{\Phi}^\intercal).$$

This equivalently leads to a linear representation of f via

$$f(x) \approx \sum_{j=1}^{m} \left(S_\theta(\sqrt{\lambda_j})\right)^{\frac{1}{2}} \phi_j(x)\beta_j, \tag{9}$$

where $\beta_j \sim \text{Normal}(0, 1)$. Thus, the function f is approximated with a finite basis function expansion (using the eigenfunctions ϕ_j of the Laplace operator), scaled by the square root of spectral density values. A key property of this approximation is that the eigenfunctions ϕ_j do not depend on the hyperparameters of the covariance function θ. Instead, the only dependence of the model on θ is through the spectral density S_θ. The eigenvalues λ_j are monotonically increasing with j and S_θ goes rapidly to zero for bounded covariance functions. Therefore, eq. (9) can be expected to be a good approximation for a finite number of m terms in the series as long as the inputs values x_i are not too close to the boundaries $-L$ and L of Ω. The computational cost of evaluating the log posterior density of univariate HSGPs scales as $O(nm + m)$, where n is the number of observations and m the number of basis functions.

The parameterization in eq. (9) is naturally in the non-centered parameterization form with independent prior distribution on β_j, which can make the posterior inference easier (see, e.g., Betancourt and Girolami, 2019). Furthermore, all dependencies on the covariance function and the hyperparameters is through the prior distribution of the regression weights β_j. The posterior distribution of the parameters $p(\boldsymbol{\beta}|\boldsymbol{y})$ is a distribution over a m-dimensional space, where m is much smaller than the number of observations n. Therefore, the parameter space is greatly reduced and this makes inference faster, especially when sampling methods are used.

3.2 Generalization to multidimensional GPs

The results from the previous section can be generalized to a multidimensional input space with compact support, $\Omega = [-L_1, L_1] \times \cdots \times [-L_D, L_D]$ and Dirichlet boundary conditions. In a D-dimensional input space, the total number of eigenfunctions and eigenvalues in the approximation is equal to the number of D-tuples, that is, possible combinations of univariate eigenfunctions over all dimensions. The number of D-tuples is given by

$$m^* = \prod_{d=1}^{D} m_d, \tag{10}$$

where m_d is the number of basis function for the dimension d. Let $\mathbb{S} \in \mathbb{N}^{m^* \times D}$ be the matrix of all those D-tuples. For example, suppose we have $D = 3$ dimensions and use $m_1 = 2$, $m_2 = 2$ and $m_3 = 3$ eigenfunctions and eigenvalues for the first, second and third dimension, respectively. Then, the number of multivariate eigenfunctions and eigenvalues is $m^* = m_1 \cdot m_2 \cdot m_3 = 12$ and the matrix $\mathbb{S} \in \mathbb{N}^{12 \times 3}$ is given by

$$\mathbb{S} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 2 \\ 1 & 1 & 1 & 2 & 2 & 2 & 1 & 1 & 1 & 2 & 2 & 2 \\ 1 & 2 & 3 & 1 & 2 & 3 & 1 & 2 & 3 & 1 & 2 & 3 \end{bmatrix}^\intercal. \tag{11}$$

Each multivariate eigenfunction $\phi_j^* : \Omega \to \mathbb{R}$ corresponds to the product of the univariate eigenfunctions whose indices corresponds to the elements of the D-tuple $\mathbb{S}_{j\cdot}$, and each multivariate eigenvalue $\boldsymbol{\lambda}_j^*$ is a D-vector with elements that are the univariate eigenvalues whose indices correspond to the elements of the D-tuple $\mathbb{S}_j$.. Thus, for $\boldsymbol{x} = \{x_d\}_{d=1}^D \in \Omega$ and $j = 1, 2, \dots, m^*$, we have

$$\boldsymbol{\lambda}_j^* = \left\{\lambda_{\mathbb{S}_{jd}}\right\}_{d=1}^D = \left\{\left(\frac{\pi \mathbb{S}_{jd}}{2L_d}\right)^2\right\}_{d=1}^D, \tag{12}$$

$$\phi_j^*(\boldsymbol{x}) = \prod_{d=1}^{D} \phi_{\mathbb{S}_{jd}}(x_d) = \prod_{d=1}^{D} \sqrt{\frac{1}{L_d}} \sin\left(\sqrt{\lambda_{\mathbb{S}_{jd}}}(x_d + L_d)\right). \tag{13}$$

The approximate covariance function is then represented as

$$k(\boldsymbol{x}, \boldsymbol{x}') \approx \sum_{j=1}^{m^*} S_\theta^*\left(\sqrt{\boldsymbol{\lambda}_j^*}\right) \phi_j^*(\boldsymbol{x})\phi_j^*(\boldsymbol{x}'), \tag{14}$$

where S_θ^* is the spectral density of the D-dimensional covariance function (see Section 2.1) as a function of $\sqrt{\boldsymbol{\lambda}_j^*}$ that denotes the element-wise square root of the vector $\boldsymbol{\lambda}_j^*$. We can now write the approximate series expansion of the multivariate function f as

$$f(\boldsymbol{x}) \approx \sum_{j=1}^{m^*} \left(S_\theta^*\left(\sqrt{\boldsymbol{\lambda}_j^*}\right)\right)^{\frac{1}{2}} \phi_j^*(\boldsymbol{x})\beta_j, \tag{15}$$

where, again, $\beta_j \sim \text{Normal}(0, 1)$. The computational cost of evaluating the log posterior density of multivariate HSGPs scales as $O(nm^* + m^*)$, where n is the number of observations and m^* is the number of multivariate basis functions.

Although this still implies linear scaling in n, the approximation is more costly than in the univariate case, as m^* is the product of the number of univariate basis functions over the input dimensions and grows exponentially with respect to the number of dimensions.

4 The accuracy of the approximation

The accuracy and speed of the HSGP model depends on several interrelated factors, most notably on the number of basis functions and on the boundary condition of the Laplace eigenfunctions. Furthermore, appropriate values for these factors will depend on the degree of non-linearity of the function to be estimated, which is in turn characterized by the length-scale of the covariance function. In this section, we analyze the effects of the number of basis functions and the boundary condition on the approximation accuracy. We present recommendations on how they should be chosen and diagnostics to check the accuracy of the obtained approximation.

Ultimately, these recommendations are based on the relationships among the number of basis functions, the boundary condition and the length-scale of the function, which depend on the particular choice of the kernel function. In this work we investigate these relationships for the square exponential and the Matérn (ν=3/2) covariance functions in the present section, and for the periodic squared exponential covariance function in Appendix B. For other kernels, the relationships will be slightly different depending on the smoothness or wigglyness of the covariance function.

4.1 Dependency on the number of basis functions and the boundary condition

As explained in Section 3, the approximation of the covariance function is a series expansion of eigenfunctions and eigenvalues of the Laplace operator in a given domain Ω, for instance in a one-dimensional input space $\Omega = [-L, L] \subset \mathbb{R}$

$$k(\tau) = \sum_{j=1}^{\infty} S_\theta(\sqrt{\lambda_j})\phi_j(\tau)\phi_j(0),$$

where L describes the boundary condition, j is the index for the eigenfunctions and eigenvalues, and $\tau = x - x'$ is the difference between two covariate values x and x' in Ω. The eigenvalues λ_j and eigenfunctions ϕ_j are given in equations (7) and (8) for the unidimensional case and in equations (12) and (13) for the multidimensional case. The number of basis functions can be truncated at some finite positive value m such that the total variation difference between the exact and approximate covariance functions is less than a predefined threshold $\varepsilon > 0$:

$$\int |k(\tau) - \sum_{j=1}^{m} S_\theta(\sqrt{\lambda_j})\phi_j(\tau)\phi_j(0)| \, \mathrm{d}\tau < \varepsilon. \tag{16}$$

This inequality can be satisfied for arbitrary small ϵ provided that L and m are sufficiently large (Solin and Särkkä, 2020, Theorem 1 and 4). The specific number of basis functions m needed depends on the degree of non-linearity of the function to be estimated, that is on its length-scale ℓ, which constitutes a hyperparameter of the GP. The approximation also depends on the boundary L (see equations (7), (8), (12) and (13)), which will affect its accuracy especially near the boundaries. As we will see later on, L will also influence the number of basis functions required in the approximation.

In this work, we choose L such that the domain $\Omega = [-L, L]$ will include the smallest interval that contains all the inputs points x_i. Without loss of generality, we can assume all data points are contained in a symmetric interval around zero. Let $S = \max_i |x_i|$, then it follows that $x_i \in [-S, S]$ for all i. We now define L as

$$L = c \cdot S, \tag{17}$$

where $S > 0$ represents the half-range of the input space, and $c \geq 1$ is the proportional extension factor. In the following, we will refer to c as the boundary factor of the approximation. The boundary factor can also be regarded as the boundary L normalized by the half-range S of the input space.

We start by illustrating how the number of basis functions m and boundary factor c influence the accuracy of the HSGP approximations individually. For this purpose, a set of noisy observations are drawn from an exact GP model with a squared exponential covariance function of length-scale $\ell = 0.3$ and marginal variance $\alpha = 1$, using input values from the zero-mean input domain with half-range $S = 1$. Several HSGP models with varying m and c are fitted to this data. In this example, the length-scale and marginal variance parameters used in the HSGPs are fixed to the true values of the data-generating model. Figures 1 and 2 illustrate the individual effects of m and c, respectively, on the posterior predictive mean and standard deviation of the estimated function as well as on the covariance function itself. For c fixed to a sufficiently large value, Figure 1 shows clearly how m affects the accuracy on the approximation for both the posterior mean or uncertainty. It is seen that if the number of basis functions m is too small, the estimated function tend to be overly smooth because the necessary high frequency components are missing. In general, the higher the degree of wigglyness of the function to be estimated, the larger number of basis functions will be required. If m is fixed to a sufficiently large value, Figure 2 shows that c affects the approximation mainly near the boundaries in mean, but along the whole domain in the

standard deviation. The approximation error for the variance tends to be bigger for the variance than for the mean.

Next, we analyze how the interaction effects between m and c affects the quality of the approximation. The length-scale and marginal variance of the covariance function will no longer be fixed but instead we compute the joint posterior distribution of the function values and the hyperparameters using the dynamic HMC algorithm implemented in Stan (Carpenter et al., 2017; Betancourt, 2017) for both the exact GP and the HSGP models. Figure 3 shows the posterior predictive mean and standard deviation of the function as well as the covariance function obtained after fitting the model for varying m and c. Figure 4 shows the root mean square error (RMSE) of the HSGP models computed against the exact GP model. Figure 5 shows the estimated length-scale and marginal variance for the exact GP model and the HSGP models. Looking at the RMSEs in Figure 4, we can conclude that the optimal choice in terms of precision and computation time for this example would be $m = 15$ basis functions and a boundary factor between $c = 1.5$ and $c = 2.5$. Further, the less conservative choice of $m = 10$ and $c = 1.5$ could also produce a sufficiently accurately approximation depending on the application. We may also come to the same conclusion by looking at the posterior predictions and covariance function plots in Figure 3. From these results, some general conclusions may be drawn:

- As c increases, m has to increase as well (and vice versa). This is consistent with the expression for the eigenvalues in eq. (7), where L appears in the denominator.
- There exists a minimum c below which an accurate approximation will never be achieved regardless of the number of basis functions m.

4.2 Theoretical evidence of linear proportionality between m, c and l

There is a clear relation between the number of basis functions m and the boundary factor c with the length-scale ℓ of the approximated function. In this Section we discuss about the existing theoretical evidence that the minimum number of basis functions m needed for an accurate approximation is linearly proportional to the the boundary factor c and inversely proportional to the lengthscale l. In next Section 4.3 we conduct an empirical experiments to prove this expected linear proportionality relationship between m, c and l and compute the specific coefficients of proportionality for every kernel.

Aki: We can also look at the mean zero-level upcrossings, which for QE is proportional to 1/l which gives as that part, and the effect of c should also be quite linear when c is not very close to 1 and m is not very small as it linearly makes the interval shorter so that the there are less zero-level upcrossings available in the basis functions. You can visualize this using `https://avehtari.github.io/casestudies/Motorcycle/motorcycle_gpcourse.html` and look how the plot of 6 basis functions change when you set c=1, c=1.5, c=2.25 (ie 1.5^2). Instead of looking at mean zero-level crossings, we can look at the spectral densities and when they go to zero. This is illustrated in the same notebook after that figure. E.g. with lengthscale l=1,l=1/2, l=1/4, l=1/8 to explain 99% of the prior variance we need 4, 8, 16, and 32 basis functions (linear in l) and to explain 80% of the prior variance we need 2, 4, 8, 16 basis functions. These spectral densities don't depend on c. When c increaes the basis functions are truncated and for example going from c=1.5 to c=2.25, basis function 9 has the flexibility basis function 6 had, so 1.5 more basis functions are needed. So we have proportionality c/l.

Gabi: The spectral densities do depend on c, see equations (9) and (7).

4.3 Empirical discovering of the functional form of the relationships between m, c and l

Figures 6 and **??** depict how these three factors interact and affect the accuracy of the HSGP approximation for a GP with square exponential covariance function and Matérn (ν=3/2) covariance function, respectively, and a single input dimension. More precisely, for a given GP model (with a squared exponential covariance function) with length-scale ℓ and given a boundary factor c, Figure 6 shows the minimum number of basis functions m required to obtain an accurate approximation in the sense of satisfying eq. (16). Similarly, Figure **??** shows the corresponding plot for the Matérn (ν=3/2) covariance function. We considered an approximation to be a sufficiently accurate when the total variation difference between the approximate and exact covariance functions, ε in eq. (16), is below 1% of the total area under the curve of the exact covariance function k such that

$$\int |k(\tau) - \tilde{k}_m(\tau)| \, \mathrm{d}\tau < 0.01 \int k(\tau) \, \mathrm{d}\tau,$$

where $\tilde{k}_m$ is the approximate covariance function with m basis functions. Alternatively, these figures can be understood as providing the minimum c that we should use for given ℓ and m. Of course, we may also read it as providing the minimum ℓ that can be approximated with high accuracy given m and c. We obtain the following main conclusions:

- As ℓ increases, m required for an accurate approximation decrease.
- The lower c, the smaller m can and ℓ must be to achieve an accurate approximation.

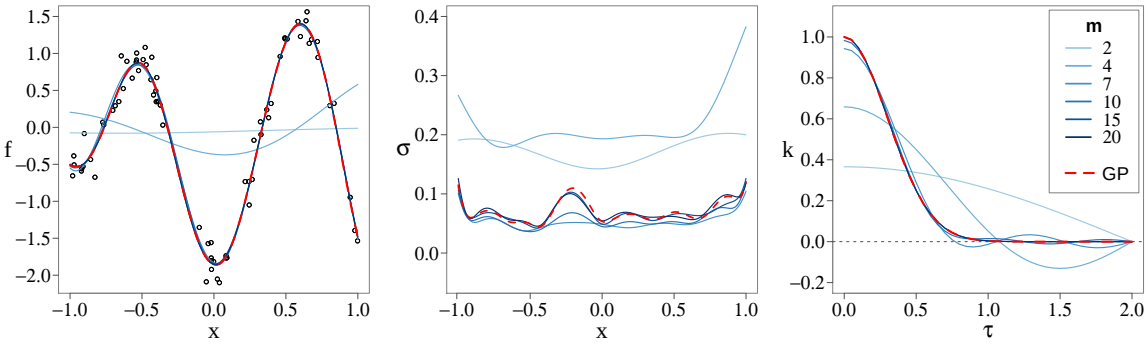

Fig. 1: Mean posterior predictive functions (left), posterior standard deviations (center), and covariance functions (right) of both the exact GP model (dashed red line) and the HSGP model for different number of basis functions m, with the boundary factor fixed to a large enough value.

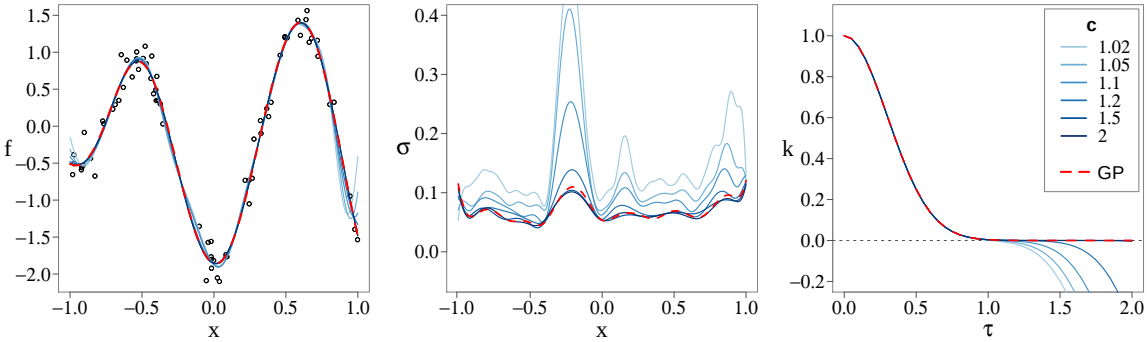

Fig. 2: Mean posterior predictive functions (left), posterior standard deviations (center), and covariance functions (right) of both the exact GP model (dashed red line) and the HSGP model for different values of the boundary factor c, with a large enough fixed number of basis functions.

– For a given ℓ there exist a minimum c under which a accurate approximation is never going to be achieved regardless of m. This fact can be seen in Figures 6 and **??** as the contour lines which represent c have an end in function of ℓ (Valid c are restricted in function of ℓ). As ℓ increases, the minimum valid c also increases.

4.3.1 Numerical equations

From the empirical observations the hypothesis of linear proportionality between m, $\frac{l}{S}$ (lenghscale l relative to the half range of the input domain S) and c has been proved. Following we show numerical functions governing these relationships for every kernel.

Squared exponential:

$$m = 1.75\,\frac{c}{l/S} \quad \Leftrightarrow \quad l/S = 1.75\,\frac{c}{m}, \tag{18}$$

with

$$c = c(l) \;\geq\; 3.2\,l/S \tag{19}$$

Matérn (ν=3/2):

$$m = 3.42\,\frac{c}{l/S} \quad \Leftrightarrow \quad l/S = 3.42\,\frac{c}{m}, \tag{20}$$

with

$$c \;\geq\; 4.5\,l/S \tag{21}$$

Matérn (ν=5/2):

$$m = 2.65\,\frac{c}{l/S} \quad \Leftrightarrow \quad l/S = 2.65\,\frac{c}{m}, \tag{22}$$

with

$$c \;\geq\; 4.1\,l/S \tag{23}$$

Figures 6 and **??** were build for a GP with a unidimensional covariance function, which result in a surface depending on three variables, m, c and ℓ. An equivalent figure for a GP model with a two-dimensional covariance function would result in a surface depending on four variables, m, c, ℓ_1 and ℓ_2, which is more difficult to be graphically represented. More precisely, in the multi-dimensional case, whether the

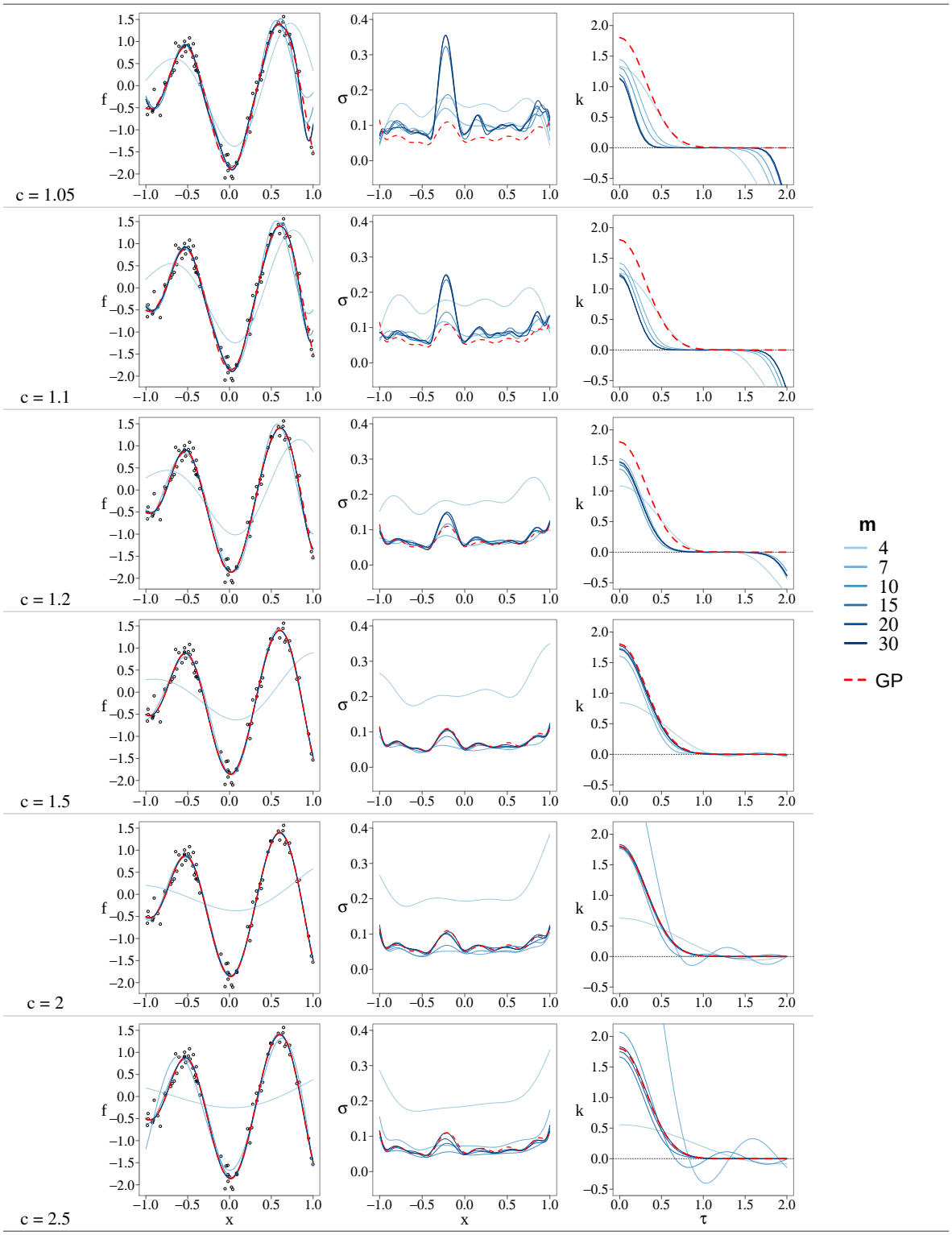

Fig. 3: Posterior mean predictive functions (left), posterior standard deviations (center) and covariance functions (right) of both the exact GP model and the HSGP model for different number of basis functions m and for different values of the boundary factor c.

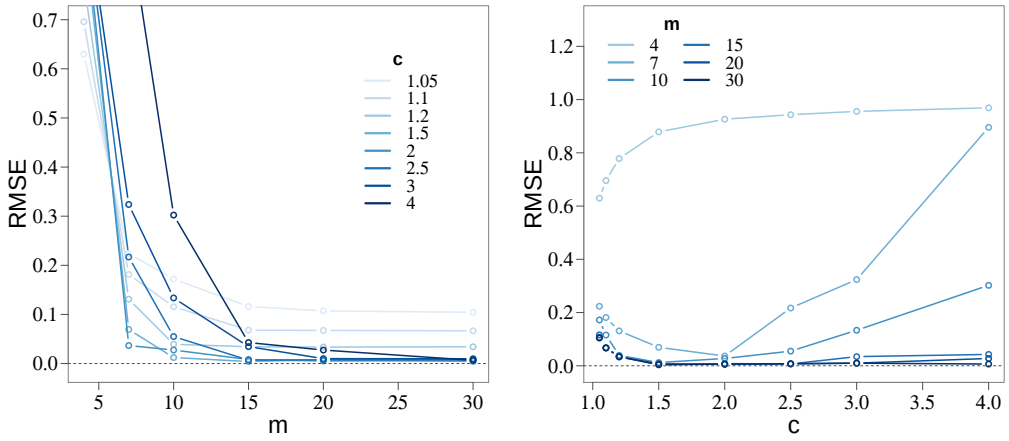

Fig. 4: Root mean square error (RMSE) of the proposed HSGP models computed against the exact GP model. RMSE versus the number of basis functions m and for different values of the boundary factor c (left). RMSE versus the boundary factor c and for different values of the number of basis functions m (right).

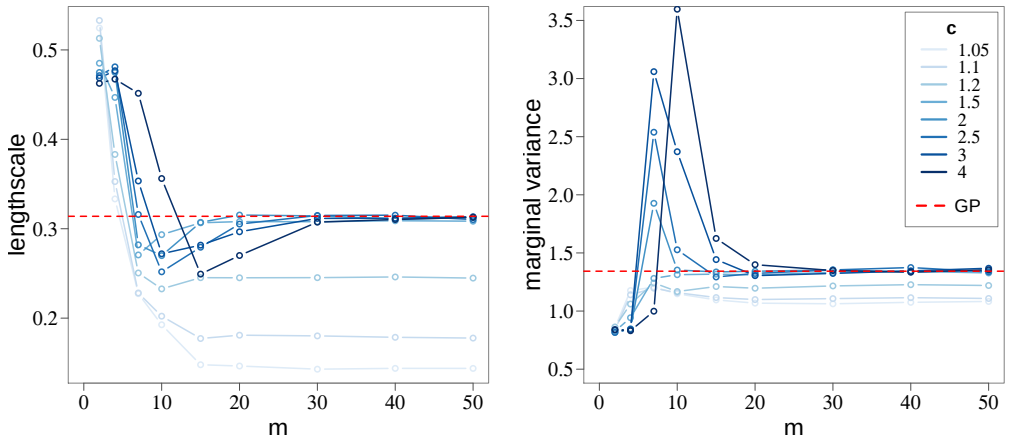

Fig. 5: Estimated length-scale (left) and marginal variance (right) parameters of both exact GP and HSGP models, plotted versus the number of basis functions m and for different values of the boundary factor c.

approximation is close enough might depend only on the ratio between wigglyness in every dimensions. For instance, in the two-dimensional case, it would depend on the ratio between ℓ_1 and ℓ_2 and could be graphically represented. Future research will focus on building useful graphs or analytical models that provide these relations in multi-dimensional cases. However, as an approximation, we can use the unidimensional GP conclusions in Figures 6 and **??** to check the accuracy by analyzing individually the different dimensions of a multidimensional GP model.

4.4 Diagnosis of the approximation

As stated above, Figures 6 and **??** provide the minimum length-scale that can be closely approximated given m and c. This information serves as a powerful diagnostic tool in determining if the obtained accuracy is acceptable. As the length-scale ℓ controls the wigglyness of the function, it strongly influences the difficulty of obtaining accurate inference about the function from the data. Basically, if the length-scale estimate is accurate, we can expect the HSGP approximation to be accurate as well.

Having obtained an estimate $\hat{\ell}$ of ℓ from the HSGP model based on prespecified m and c, we can check whether or not

$\hat{\ell}$ exceeds the minimum length-scale provided in Figure 6 or **??** (depending on which kernel is used). If $\hat{\ell}$ exceeds this recommended minimum length-scale, the approximation is assumed to be good. If, however, $\hat{\ell}$ does not exceed recommended minimum length-scale, the approximation may be inaccurate and m should be increased or c decreased. We may also use this diagnostic in an iterative procedure by starting from some initial guess of ℓ and initial values for m and c, and if the estimated $\hat{\ell}$ is below the minimum length-scale, repeat the process while increasing m or decreasing c. As mentioned earlier, c cannot be decreased too much as the lowest useful value of c is restricted by the length-scale. Thus, increasing m may usually the preferred approach.

If we look back to the conclusions drawn from Figures 4 and 5, where $m = 10$ basis functions and a boundary factor of $c = 1.5$ were sufficient to obtain an accurate approximation of a function with $\ell = 0.3$, we can recognize that these conclusions also matches those obtained from Figure **??**.

4.4.1 User guide with the steps to perform diagnosis

Assumption of the diagnosis tool: Under inaccurate HSGP approximation, the estimated lengthscale $\hat{l}$ is smaller than the exact GP lengthscale estimate l. Thus the diagnostic that determines is the approximation is sufficiently accurate is:

$$\hat{l} \pm 0.02 \geq l \tag{24}$$

User-guide with the steps to perform diagnosis:

Step 1. Make a first guess on the lengthscale l_1 of the function to be learned. We recommend a large lengthscale, around 1.

(Initial iteration, $i = 1$)

Step 2. Obtain the valid minimum boundary factor c_1 determined by l_1 by using (19).

Step 3. Obtain the valid mimimum number of basis functions m_1 determined by l_1 and c_1 by using (18). Notice that l_1 can also be read as the minimum lengthscale that can be accurately fitted determined by m_1 and c_1 by using (18).

Step 4. Fit the HSGP model and assess residuals: compute *rmse** (Root mean square error), R^2 (Coefficient of variation) and *elpd* (Expected log predicitve density).

Step 5. Check the diagnostic of whether $\hat{l_1} \pm 0.02 \geq l_1$ (24).

(a) If the diagnostic is TRUE, the HSGP model approximation must be close to be sufficiently accurate. Then user can continue with step 6.

(Next iterations, $i > 1$)

Step 6. Set $m_i = m_{i-1} + 10$.

Step 7. Obtain the valid minimum boundary factor c_i by using (19), with $c(\hat{l}_{i-1})$.

Step 8. Obtain the mimimum lengthscale l_i that can be accurately fitted determined by m_i and c_i by using (18).

Step 9. Fit the HSGP model and

Step 9.1. Check whether $\hat{l}_i \pm 0.02 \geq l_i$ (24),

Step 9.2. Check stability in $\hat{l}_i$, *rmse*, R^2 and *elpd* relative to previous iteration.

Step 10. If the verifications in steps 9.1 and 9.2 are TRUE, the HSGP model approximation should be sufficiently accurate, and diagnosis ends here. Otherwise, repeat steps 6-10 and update parameters.

(b) If the diagnostic is FALSE, the HSGP model approximation can not be sufficiently accurate. Then user has to move to step 11.

Step 11. Set $l_i = \hat{l}_{i-1}$.

Step 12. Repeat steps 2-5 and update parameters.

4.5 Comparing length-scale estimates

In this example, we make a comparison of the length-scale estimates obtained from the exact GP and HSGP models. We also have a look at those recommended minimum length-scales provided by Figure **??**. For this analysis, we will use various datasets consisting of noisy draws from a GP prior model with a squared exponential covariance function and varying length-scale values. Different values of the number of basis functions m are used when estimating the HSGP models, and the boundary factor c is set to a valid and optimum value in every case.

Figure 7 shows the posterior predictions of both exact GP and HSGP models fitted to those datasets. The length-scale estimates as obtained by exact GP and HSGP models are depicted in Figure 8. As noted previously, an accurate estimate of the length-scale can be a good indicator of a close approximation of the HSGP model to the exact GP model. Further, Figure 9 shows the root mean square error (RMSE) of the HSGP models, computed against the exact GP models, as a function of the length-scale and number of basis functions.

Comparing the accuracy of the length-scale in Figure 8 to the RMSE in Figure 9, we see that they agree closely with each other for medium length-scales. That is, a good estimation of the length-scale implies a small RMSE. This is no longer true for very small or large length-scales. In small length-scales, even very small inaccuracies may have a strong influence on the posteriors predictions and thus on the RMSE. In large

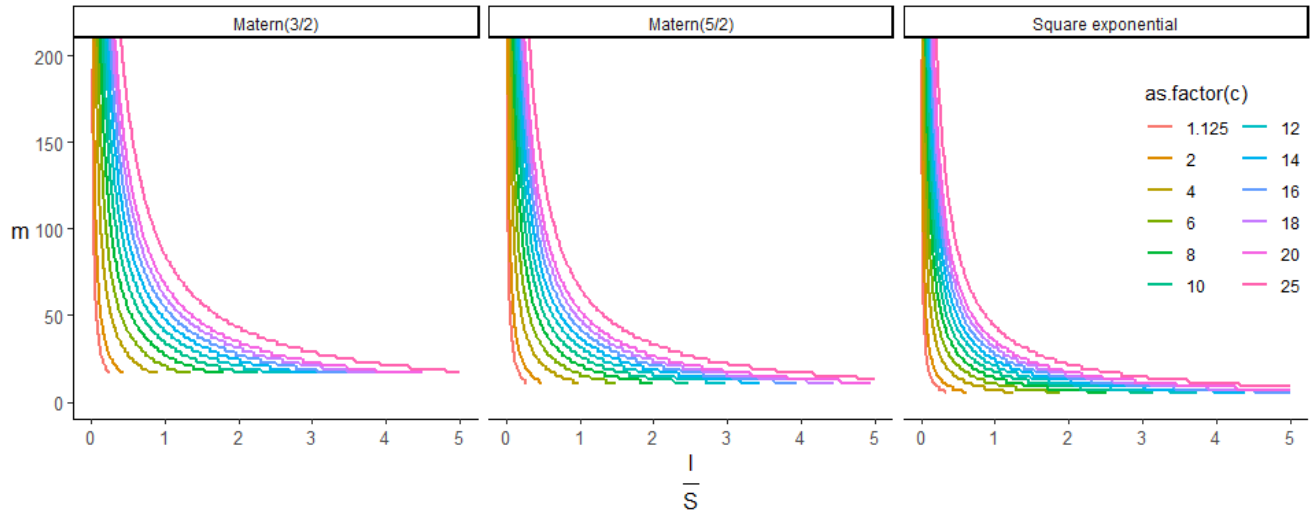

Fig. 6: Relation among the minimum number of basis functions m, the boundary factor c ($c = \frac{L}{S}$) and the length-scale normalized by the half-range of the data ($\frac{\ell}{S}$), for the square exponential, Matérn (ν=3/2) and Matérn (ν=5/2) covariance functions.

length-scales, larger inaccuracies change the posterior predictions only little and may thus not yield large RMSEs. The dashed black line in Figure 8 represents the minimum length-scale that can be closely approximated under the given condition, according to the results presented in Figure **??**. We observe that whenever the estimated length-scale exceeds the minimally estimable length-scale, the RMSE of the posterior predictions is small (see Figure 9). Conversely, when the estimated length-scale is smaller than the minimally estimable one, the RMSE becomes very large.

5 Case studies

In this section, we will present several simulated and real case studies in which we apply the developed HSGP models. More case studies are presented in the online supplemental materials.

5.1 Simulated data for a univariate function

In this experiment, we analyze a synthetic dataset with $n = 250$ observations, where the true data generating process is a Gaussian process with additive noise. The data points are simulated from the model $y_i = f(x_i) + \epsilon_i$, where f is a sample from a Gaussian process using the Matérn (ν=3/2) covariance function with marginal variance $\alpha = 1$ and length-scale $\ell = 0.15$ at inputs values $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ with $x_i \in [-1, 1]$. ϵ_i is additive Gaussian noise with standard deviation $\sigma = 0.2$. We split the dataset into three parts: 155 data points are used for fitting the model (training set), 45 data ponts are used for the interpolation test set, and the remaining 50 data points are used for the extrapolation test set.

The exact GP model for fitting and predicting this simulated dataset $\boldsymbol{y}$ can be written as follows,

$$\begin{aligned}\boldsymbol{y} &= \boldsymbol{f} + \boldsymbol{\epsilon}\\ \boldsymbol{\epsilon} &\sim \text{Normal}(\boldsymbol{0}, \sigma^2 \boldsymbol{I})\\ f(x) &\sim \mathcal{GP}(0, k(x, x', \theta)),\end{aligned}$$

where $\boldsymbol{f} = \{f(x_i)\}_{i=1}^n$ represents the underlying function at the input values x_i, and $\boldsymbol{\epsilon}$ is the Gaussian noise term with variance σ^2, with $\boldsymbol{I}$ representing the identity matrix. The function $f : \mathbb{R} \to \mathbb{R}$ is a GP prior with the Matérn (ν=3/2) covariance function k, which depends on the inputs $\boldsymbol{x}$ and hyperparameters $\theta = \{\alpha, \ell\}$. The hyperparameters α and ℓ represent the marginal variance and length-scale, respectively, of the GP process. Saying that the function $f(\cdot)$ follows a GP model is equivalent to say that $\boldsymbol{f}$ is multivariate Gaussian distributed with covariance matrix $\boldsymbol{K}$, where $K_{ij} = k(x_i, x_j, \theta)$, with $i, j = 1, 2, \ldots, n$.

A more computationally efficient formulation of a GP model with Gaussian likelihood, and for probabilistic inference using MCMC sampling methods, would be its marginalized form,

$$\boldsymbol{y} \sim \text{Normal}(\boldsymbol{0}, \boldsymbol{K} + \sigma^2 \boldsymbol{I}),$$

where the function values $\boldsymbol{f}$ have been integrated out, yielding a lower-dimensional parameter space over which to do inference, reducing the time of computation and improving the sampling and the effective number of samples.

In the HSGP model, the latent function values $f(x)$ are approximated as in eq. (9), with the Matérn (ν=3/2) spectral

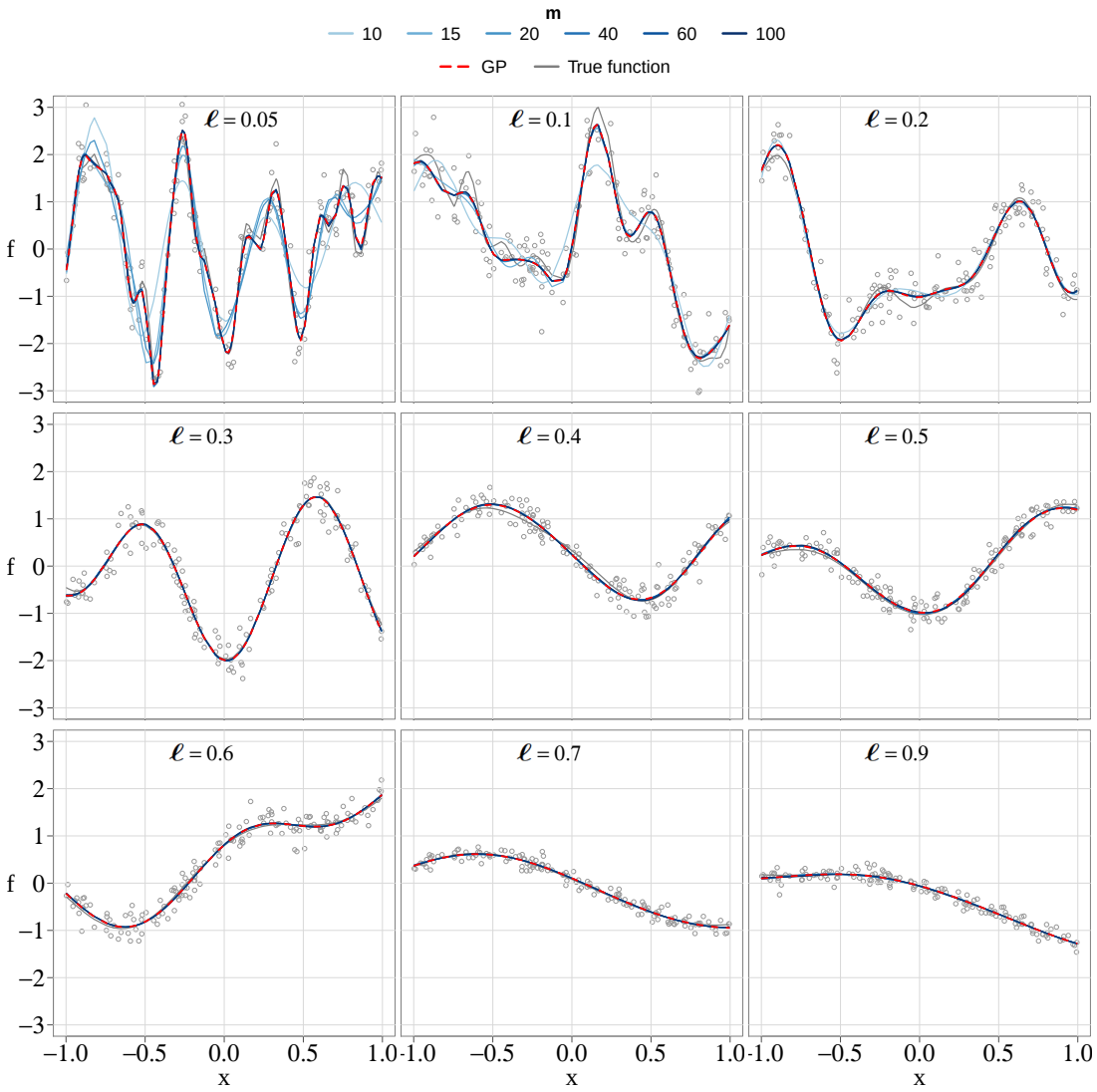

Fig. 7: Mean posterior predictions of both exact GP and HSGP models, fitted over various datasets drawn from square exponential GP models with different characteristic length-scales (ℓ) and same marginal variance (α) as the data-generating functions (*true function*).

density S as in eq. (3), and eigenvalues λ_j and eigenfunctions ϕ_j as in equations (7) and (8), respectively. In order to do model comparison, in addition to the exact GP model and HSGP model, a spline-based model is also fitted using the thin plate regression spline approach by Wood (2003) and implemented in the R-package *mgcv* (Wood, 2011). A Bayesian approach is used to fit this spline model using the R-package *brms* (Bürkner, 2017).

The joint posterior parameter distributions are estimated by sampling using the dynamic HMC algorithm implemented in Stan (Carpenter et al., 2017; Betancourt, 2017). A $\text{Gamma}(1, 1)$ prior distribution has been used for both observation noise σ and covariance function marginal variance α, and a $\text{Gamma}(3.75, 25)$ prior distribution for length-scale ℓ. We use the same prior distributions for the exact GP model as for the HSGP models.

Figure 10 shows the posteriors predictive distributions of the three models, the exact GP, the HSGP with $m = 60$ basis functions and boundary factor $c = 1.2$ ($L = c \cdot 1 = 1.2$; see eq. (17)), and the spline model with 80 knots. The true data-generating function and the noisy observations are also plotted. The sample observations are plotted as circles and the out-of-sample or test data are plotted as crosses. The test data located at the extremes of the plot are used for assessing model extrapolation, and the test data located in the middle are used for assessing model interpolation. The posteriors

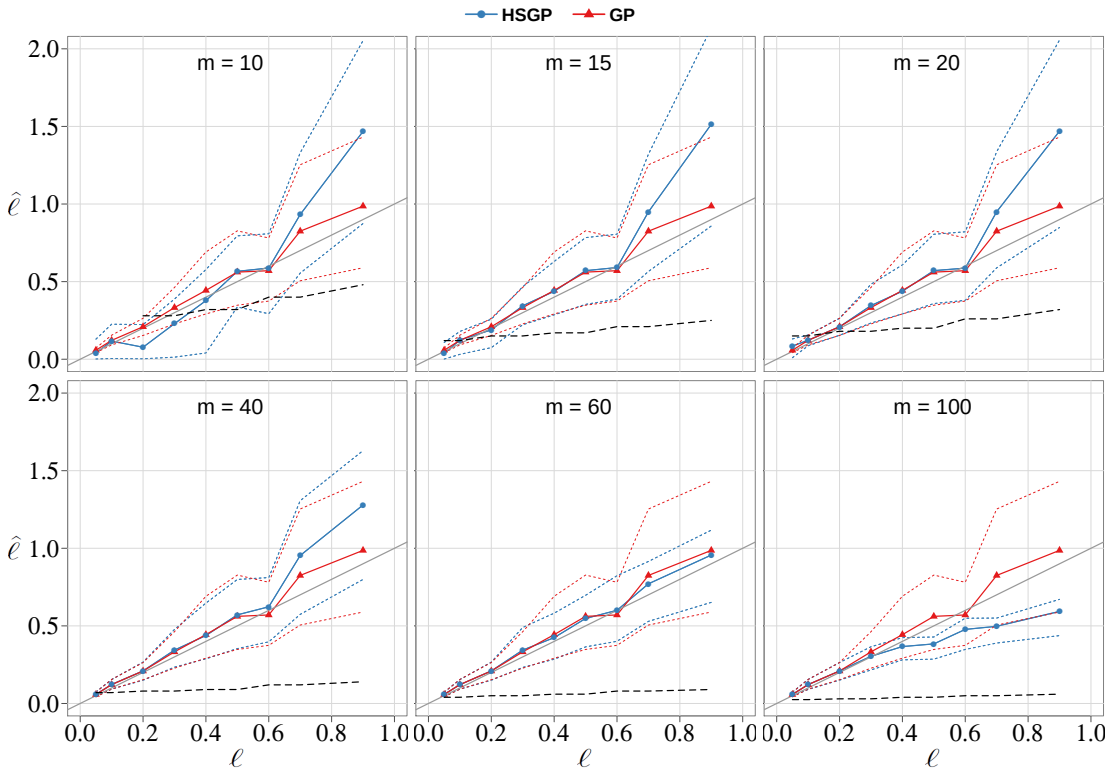

Fig. 8: Data-generating functional length-scales (ℓ), of the various datasets illustrated in Figure 7, versus the corresponding length-scale estimates ($\hat{\ell}$) from the exact GP and HSGP models. 95% confident intervals of the length-scale estimates are plotted as dot lines. The different plots represent the use of different number of basis functions m in the HSGP model. The dashed black line represents the recommended minimum length-scales provided by Figure **??** that can be closely approximated by the HSGP model in every case.

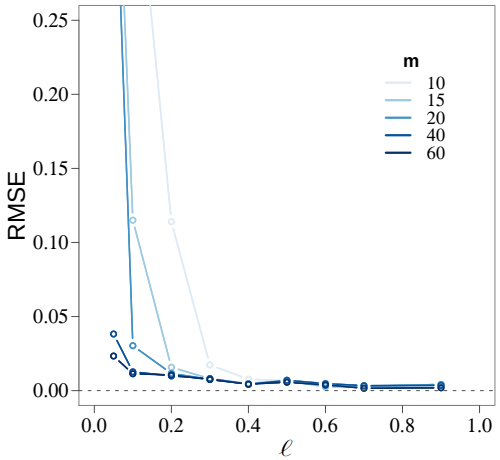

Fig. 9: RMSE of the HSGP models with different number of basis functions m, for the various datasets with different wiggly effects (ℓ).

of the three models, exact GP, HSGP and spline, are similar in the interpolation regions of the input space. However, when extrapolating the spline model solution clearly differs from the exact GP and HSGP models as well as the actual observations.

In order to assess the performance of the models as a function of the number of basis functions and number of knots, different models with different number of basis functions for the HSGP model, and different number of knots for the spline model, have been fitted. Figure 11 shows the standardized root mean squared error (SRMSE) for interpolation and extrapolating data as a function of the number of basis functions and knots. The SRMSE is computed against the data-generating model. From Figures 10 and 11, it is seen that the HSGP method yields a good approximation of the exact GP model for both interpolation and extrapolation. However, the spline model does not extrapolate data properly. Both models show roughly similar interpolating performance.

Next, we demonstrate how to apply the diagnostic for approximation quality as a function of m and c as proposed in Section 4.1. Figure 12 depicts the diagnosis applied to case study. In the first iteration, we choose $m = 20$ and $c = 1.5$, which turns out to be an invalid combination of m and c, as $c = 1.5$ is too small for $m = 20$ or $m = 20$ is too small for $c = 1.5$, as can be seen in the corresponding plot of Figure 12. In the second iteration, m is increased to $m = 30$, the HSGP estimates a length-scale of $\hat{\ell} = 0.12$, and the minimum length-scale that can be accurately approximated, for

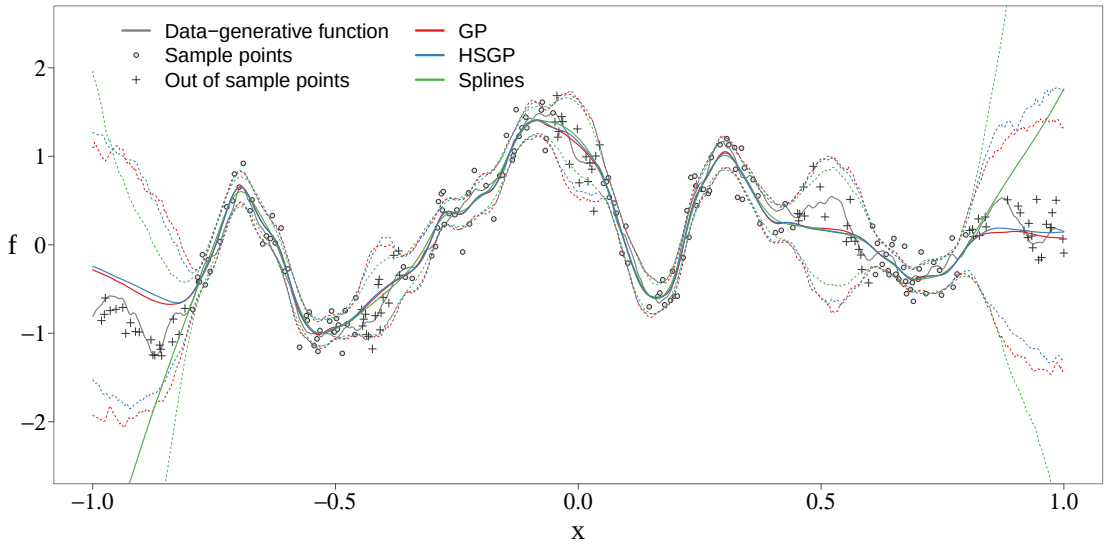

Fig. 10: Posterior predictive means of the proposed HSGP model, the exact GP model, and the spline model. 95% credible intervals are plotted as dashed lines.

those m and c, generated by the model of relationships in Figure **??** is $\ell^* = 0.19$. As $\hat{\ell} < \ell^*$, the approximation is diagnosed to be inaccurate (see Figure 11-left where the interpolation error of HSGP is higher than that of GP). In the third iteration, m is increased to $m = 40$, the HSGP model estimates a length-scale of $\hat{\ell} = 0.14$, and the model of relationships suggests a minimum length-scale of $\ell^* = 0.15$. As $\hat{\ell}$ is still lower than ℓ^*, the approximation is diagnosed to be inaccurate, but since $\hat{\ell}$ is quite close to ℓ^*, the error is expected to be small (see errors in Figure 11-left). In the fourth iteration, m is increased up $m = 60$, the estimated length-scale is $\hat{\ell} = 0.14$ and the minimum length-scale generated by the model of relationships is $\ell^* = 0.11$, and the approximation is finally diagnosed to be accurate as $\hat{\ell} > \ell^*$.

Figure 13 shows computational times, in seconds per iteration (iteration of the HMC sampling method), as a function of the number of basis functions m, for the HSGP model, and knots, for the spline model. The HSGP model is on average roughly 400 times faster than the exact GP and 10 times faster than the spline model for this particular model and data. Also, it is seen that the computation time increases slowly as a function of the number of basis functions.

The Stan model code for the exact GP, the approximate GP and the spline models of this case study can be found online at `https://github.com/gabriuma/basis_functions_approach_to_GP/tree/master/Paper/Case-study_1D-Simulated-data`.

In the online supplemental material, additional case studies are presented. From those examples, it can be seen how the computation time of the HSGP model increases rapidly with the number of input dimensions (D) since the number of basis functions in the approximation increases exponentially with D (see eq. (10)). Even though, for a bivariate input space, the computation time increases significantly with D, the HSGP model works significantly faster than the exact GP for most of the non-linear $2D$ functions (even highly wiggly functions; see Figures B.3-right and C.3 in the online material). For moderate sized datasets, HSGPs tend to be slower than exact GPs for $D > 3$ with a relatively low number of basis functions ($m \gtrsim 5$), as well as even for $D = 3$ with a moderate high number of basis functions ($m \gtrsim 20$; see Figure C.3 in the online material). However, the HSGP method will be computationally faster than the exact GP for larger datasets due the cubic scaling of the exact GP. In all of the investigated cases, choosing the optimal boundary factor in the HSGP approximation reduces the number of required basis functions noticeably (see Figures A.3, B.3-left and C.2 in the online material) and therefore also reduces computational time drastically in particular in multivariate input spaces.

Roughly similar or even worse behavior was found for splines where serious difficulties with computation time were encountered in building spline models for $D = 3$ and with more than 10 knots, or even for $D = 2$ and more than 40 knots (see Figures B.3-right and C.3 in the online material).

5.2 Birthday data

This example is an analysis of patterns in birthday frequencies in a dataset containing records of all births in the United States on each day during the period 1969–1988. The model decomposes the number of births along all the period in

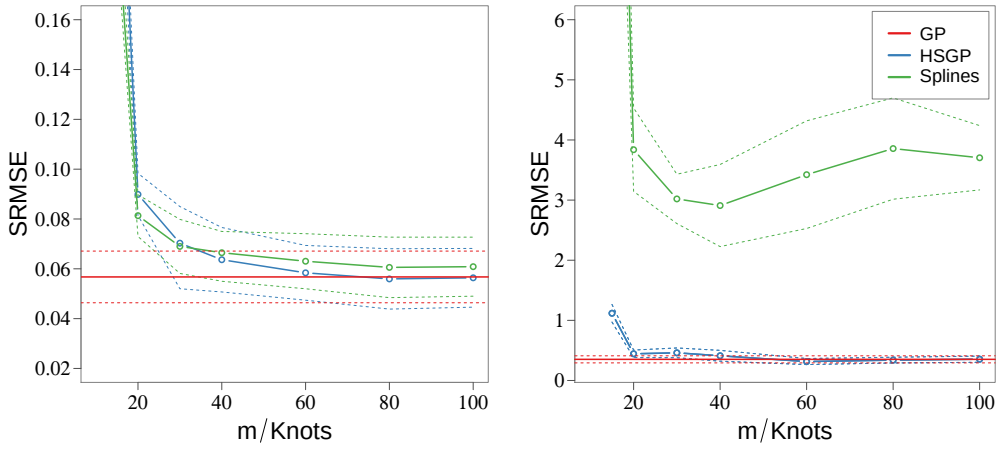

Fig. 11: Standardized root mean square error (SRMSE) of the different methods against the data-generating function. SRMSE for interpolation (left) and SRMSE for extrapolation (right). The standard deviation of the mean of the SRMSE is plotted as dashed lines.

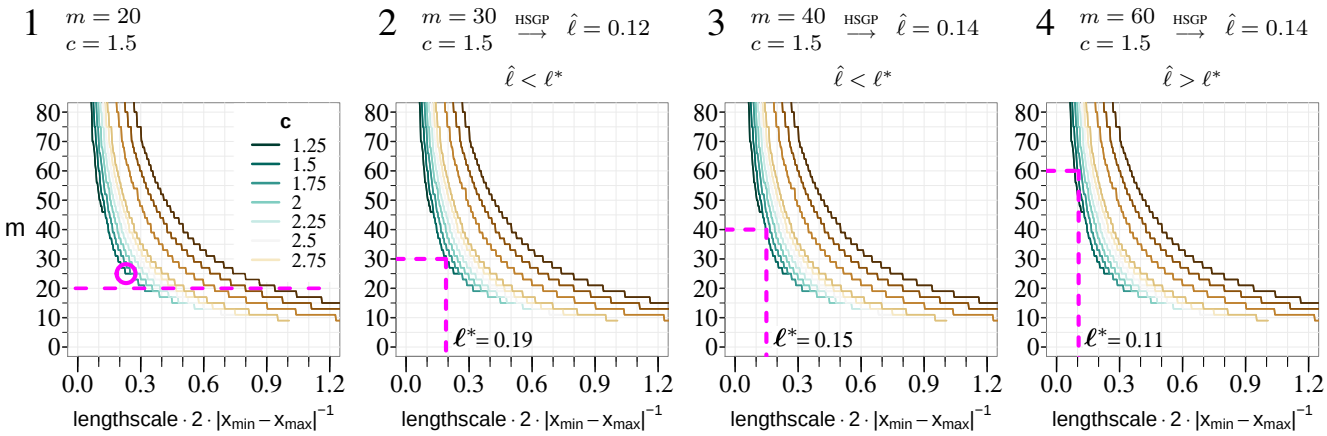

Fig. 12: Iterative diagnosis of the approximation as a function of the number of basis functions m and boundary factor c. The iterative steps go from the left side to the right side of the figure.

longer-term trend effects, patterns during the year, day-of-week effects, and special days effects. The special days effects cover patterns such as possible fewer births on Halloween, Christmas or new year, and excess of births on Valentine's Day or the days after Christmas (due, presumably, to choices involved in scheduled deliveries, along with decisions of whether to induce a birth for health reasons). Gelman et al. (2013) presented an analysis using exact GP and maximum a posteriori inference. As the total number of days within the period is $T = 7305$ ($t = 1, 2, \ldots, T$), a full Bayesian inference with MCMC for a exact GP model is memory and time consuming. We will use the HSGP method as well as the low-rank GP model with a periodic covariance function described in Appendix B which is based on expanding the periodic covariance function into a series of stochastic resonators (Solin and Särkkä, 2014).

Let y_t denote the number of births on the t'th day. The observation model is a normal distribution with mean function $\mu(t)$ and noise variance σ^2,

$$y_t \sim \text{Normal}(\mu(t), \sigma^2).$$

The mean function $\mu(t)$ will be defined as an additive model in the form

$$\mu(t) = f_1(t) + f_2(t) + f_3(t) + f_4(t). \tag{25}$$

The component $f_1(t)$ represents the long-term trends modeled by a GP with squared exponential covariance function,

$$f_1(t) \sim \mathcal{GP}(0, k_1), \qquad k_1(t, t') = \alpha_1 \exp\left(-\frac{1}{2}\frac{(t-t')^2}{\ell_1^2}\right),$$

which means the function values $\boldsymbol{f}_1 = \{f_1(t)\}_{t=1}^T$ are multivariate Gaussian distributed with covariance matrix $\boldsymbol{K}_1$,

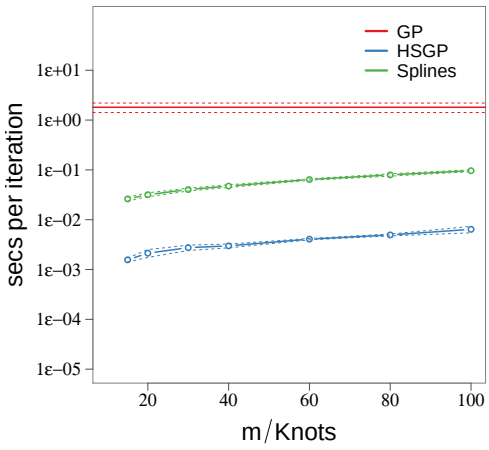

Fig. 13: Computational time (y-axis), in seconds per iteration (iteration of the HMC sampling method), as a function of the number of basis functions m, for the HSGP model, and knots, for the spline model. The y-axis is on a logarithmic scale. The standard deviation of the computational time is plotted as dashed lines.

where $K_{1_{t,s}} = k_1(t,s)$, with $t, s = 1, \ldots, T$. α_1 and ℓ_1 represent the marginal variance and length-scale, respectively, of this GP prior component. The component $f_2(t)$ represents the yearly smooth seasonal pattern, using a periodic squared exponential covariance function (with period 365.25 to match the average length of the year) in a GP model,

$$f_2(t) \sim \mathcal{GP}(0, k_2),$$
$$k_2(t,t') = \alpha_2 \exp\left(-\frac{2\,\sin^2(\pi(t-t')/365.25}{\ell_2^2}\right).$$

The component $f_3(t)$ represents the weekly smooth pattern using a periodic squared exponential covariance function (with period 7 of length of the week) in a GP model,

$$f_3(t) \sim \mathcal{GP}(0, k_3),$$
$$k_3(t,t') = \alpha_3 \exp\left(-\frac{2\,\sin^2(\pi(t-t')/7}{\ell_3^2}\right).$$

The component $f_4(t)$ represents the special days effects, modeled as a horse-shoe prior model (Carvalho et al., 2010; Piironen and Vehtari, 2017b):

$$f_4(t) \sim \text{Normal}(0, \lambda_t^2\tau^2), \qquad \lambda_t^2 \sim \mathcal{C}^+(0,1).$$

A horse-shoe prior allows for sparse distributed effects. Its global parameter τ pulls all the weights (effects) globally towards zero, while the thick half-Cauchy tails for the local scales λ_t allow some of the weights to escape the shrinkage. Different levels of sparsity can be accommodated by changing the value of τ: for large τ all the variables have very diffuse prior distributions with very little shrinkage towards zero, but letting $\tau \to 0$ will shrink all the weights $f_4(t)$ to zero (Piironen and Vehtari, 2017a).

The component $f_1(t)$ will be approximated using the HSGP model and the function values $f_1(t)$ are approximated as in eq. (9), with the squared exponential spectral density S as in eq. (1), and eigenvalues λ_j and eigenfunctions ϕ_j as in equations (7) and (8). We use $m = 30$ basis functions and a boundary factor $c = 1.5$. The length-scale estimate $\hat{\ell}_1$, for this component, normalized by half of the range of the input x_1, is bigger than the minimum length-scale reported by Figure **??** as a function of m and c. This means that the chosen number of basis functions and the boundary factor are suitable values for modeling the input effects sufficiently accurate.

The year effects $f_2(t)$ and week effects $f_3(t)$ use a periodic covariance function and thus do no fit under the main framework of the HSGP approximation covered in this paper. However, they do have a representation based on expanding periodic covariance functions into a series of stochastic resonators (Appendix B). Thus, the functions $f_2(t)$ and $f_3(t)$ are approximated as in eq. (B.7), with variance coefficients $\tilde{q}_j^2$ as in eq. (B.5). We use $J = 10$ cosine terms. The length-scale estimates $\hat{\ell}_2$ and $\hat{\ell}_3$ for the GP components $f_2(t)$ and $f_3(t)$, respectively, are bigger than the minimum length-scale reported by Figure B.1 as function of the number of cosine terms J, which means that the approximations are good.

Figure 14 shows the posterior means of the long-term trend $f_1(t)$ and yearly pattern $f_2(t)$ for the whole period, jointly with the observed data. Figure 15 shows the model for one year (1972) only. In this figure, the special days effects $f_4(t)$ in the year can be clearly represented. The posterior means of the the function $\mu(t)$ and the components $f_1(t)$ (long-term trend) and $f_2(t)$ (year pattern) are also plotted in this Figure 15. Figure 16 show the process in the month of January of 1972 only, where the week pattern $f_3(t)$ can be clearly represented. The mean of the the function $\mu(t)$ and components $f_1(t)$ (long-term trend), $f_2(t)$ (year pattern) and $f_4(t)$ (special-days effects) are also plotted in this Figure 16.

The Stan model code for the approximate GP model of this case study can be found online at `https://github.com/gabriuma/basis_functions_approach_to_GP/tree/master/Paper/Case-study_Birthday-data`.

5.3 Leukemia data

The next example presents a survival analysis in acute myeloid leukemia (AML) in adults, with data recorded between 1982 and 1998 in the North West Leukemia Register in the United Kingdom. The data set consists of survival and censoring times t_i and censoring indicator z_i (0 for observed and 1 for

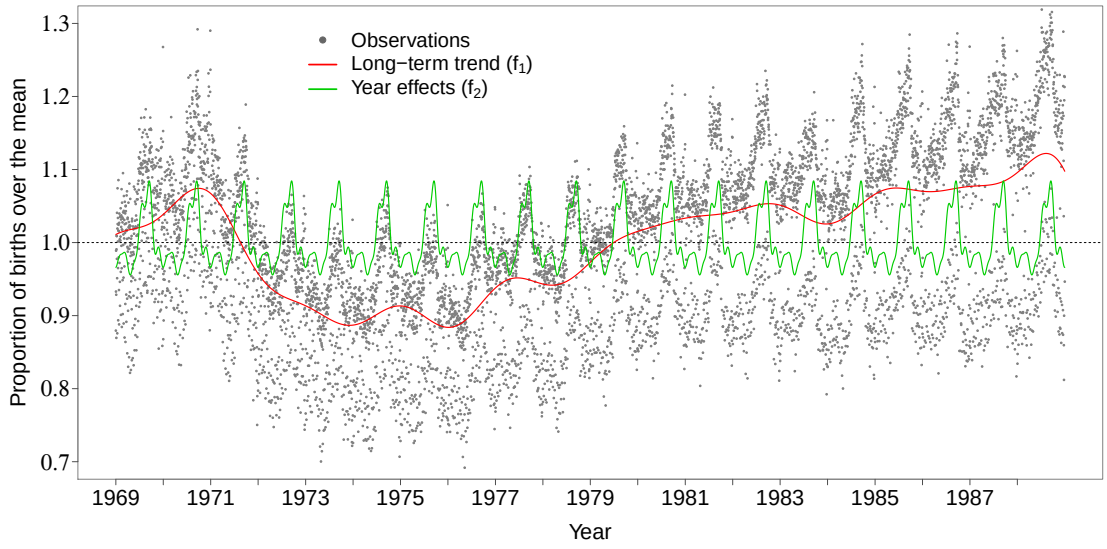

Fig. 14: Posterior means of the long-term trend ($f_1(\cdot)$) and year effects pattern ($f_2(\cdot)$) for the whole series.

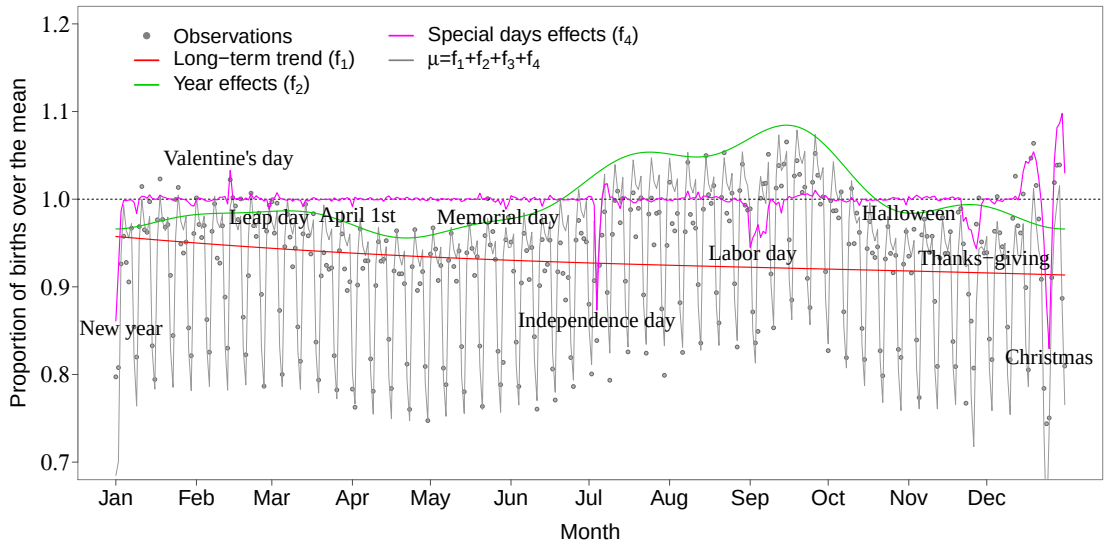

Fig. 15: Posterior means of the function $\mu(\cdot)$ for the year 1972 of the series. The special days effects pattern ($f_4(\cdot)$) in the year is also represented, as well as the long-term trend ($f_1(\cdot)$) and year effects pattern ($f_2(\cdot)$).

censored) for $n = 1043$ cases ($i = 1, \ldots, n$). About 16% of cases were censored. Predictors are *age* (x_1), *sex* (x_2), *white blood cell* (WBC) (x_3) count at diagnosis with 1 unit = $50 \times 109/L$, and the *Townsend deprivation index* (TDI) (x_4) which is a measure of deprivation for district of residence. We denote $\boldsymbol{x}_i = (x_{i1}, x_{i2}, x_{i3}, x_{i4}) \in \mathbb{R}^4$ as the vector of predictor values for observation i.

As the WBC predictor values were strictly positive and highly skewed, a logarithm transformation is used. Continuous predictors were normalized to have zero mean and unit standard deviation. We assume a log-normal observation model for the observed survival time, t_i, with a function of the predictors, $f(\boldsymbol{x}_i) : \mathbb{R}^4 \to \mathbb{R}$, as the location parameter, and σ as the Gaussian noise:

$$p(t_i \mid f_i) = \text{LogNormal}(t_i \mid f(\boldsymbol{x}_i), \sigma^2).$$

We do not have a full observation model, as we do not have a model for the censoring process. We use the complementary cumulative log-normal probability distribution for the

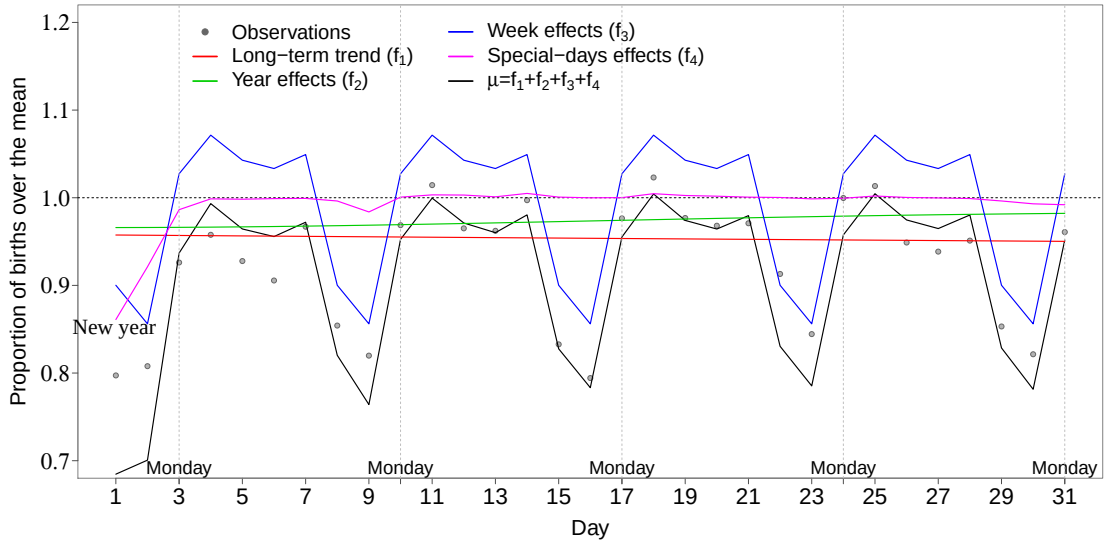

Fig. 16: Posterior means of the function $\mu(\cdot)$ for the month of January of 1972. The week effects pattern ($f_3(\cdot)$) in the month is also represented, as well as the long-term trend ($f_1(\cdot)$), year effects pattern ($f_2(\cdot)$) and special days effects pattern ($f_4(\cdot)$).

censored data conditionally on the censoring time t_i:

$$p(y_i > t_i \mid f) = \int_{t_i}^{\infty} \text{LogNormal}(y_i \mid f(\boldsymbol{x}_i), \sigma^2)\, \mathrm{d}y_i$$
$$= 1 - \Phi\left(\frac{\log(y_i) - f(\boldsymbol{x}_i)}{\sigma}\right),$$

where $y_i > t_i$ denotes the unobserved survival time. The latent function $f(\cdot)$ is modeled as a Gaussian process, centered around a linear model of the predictors $\boldsymbol{x}$, and with a squared exponential covariance function k. Due to the predictor *sex* (x_2) being a categorical variable ('1' for female and '2' for male), we apply indicator variable coding for the GP functions, in a similar way such coding is applied in linear models (Gelman et al., 2020). The latent function $f(\boldsymbol{x})$, besides of being centered around a linear model, is composed of a general mean GP function, $h(\boldsymbol{x})$, defined for all observations, plus a second GP function, $g(\boldsymbol{x})$, that only applies to one of the predictor levels ('male' in this case) and is set to zero otherwise:

$$h(\boldsymbol{x}) \sim \mathcal{GP}\left(\mathbf{0},\, k(\boldsymbol{x}, \boldsymbol{x}', \theta_0)\right),$$
$$g(\boldsymbol{x}) \sim \mathcal{GP}\left(\mathbf{0},\, k(\boldsymbol{x}, \boldsymbol{x}', \theta_1)\right),$$
$$f(\boldsymbol{x}) = c + \boldsymbol{\beta}\boldsymbol{x} + h(\boldsymbol{x}) + \mathbb{I}[x_2 = 2]\, g(\boldsymbol{x}),$$

where $\mathbb{I}[\cdot]$ is an indicator function. Above, c and $\boldsymbol{\beta}$ are the intercept and vector of coefficients, respectively, of the linear model. θ_0 contains the hyperparameters α_0 and ℓ_0 which are the marginal variance and length-scale of the general mean GP function, and θ_1 contains the hyperparameters α_1 and ℓ_1 which are the marginal variance and length-scale, respectively, of a GP function specific to the male sex ($x_2 = 2$). Scalar length-scales, l_0 and l_1, are used in both multivariate covariance functions, assuming isotropic functions.

Using the HSGP approximation, the functions $h(\boldsymbol{x})$ and $g(\boldsymbol{x})$ are approximated as in eq. (15), with the D-dimensional (with a scalar length-scale) squared exponential spectral density S as in eq. (1), and the multivariate eigenfunctions ϕ_j and the D-vector of eigenvalues $\boldsymbol{\lambda}_j$ as in equations (13) and (12), respectively.

Figure 17 shows estimated conditional functions of each predictor with all others fixed to their mean values. These posterior estimates correspond to the HSGP model with $m = 10$ basis functions and $c = 3$ boundary factor. There are clear non-linear patterns and the right bottom subplot also shows that the conditional function associated with WBC has an interaction with TDI. Figure 18 shows the expected log predictive density (ELPD; Vehtari and Ojanen, 2012; Vehtari et al., 2017) and time of computation as function of the number of univariate basis functions m ($m^* = m^D$ in eq. (15)) and boundary factor c. As the functions are smooth, a few number of basis functions and a large boundary factor are required to obtain a good approximation (Figure 18-left); Small boundary factors are not appropriate for models for large length-scales, as can be seen in Figure **??**. Increasing the boundary factor also significantly increases the time of computation (Figure 18-right). With a moderate number of univariate basis functions ($m = 15$), the HSGP model becomes slower than the exact GP model, in this specific application with 3 input variables, as the total number of multivariate basis functions becomes $15^3 = 3375$ and is therefore quite high.

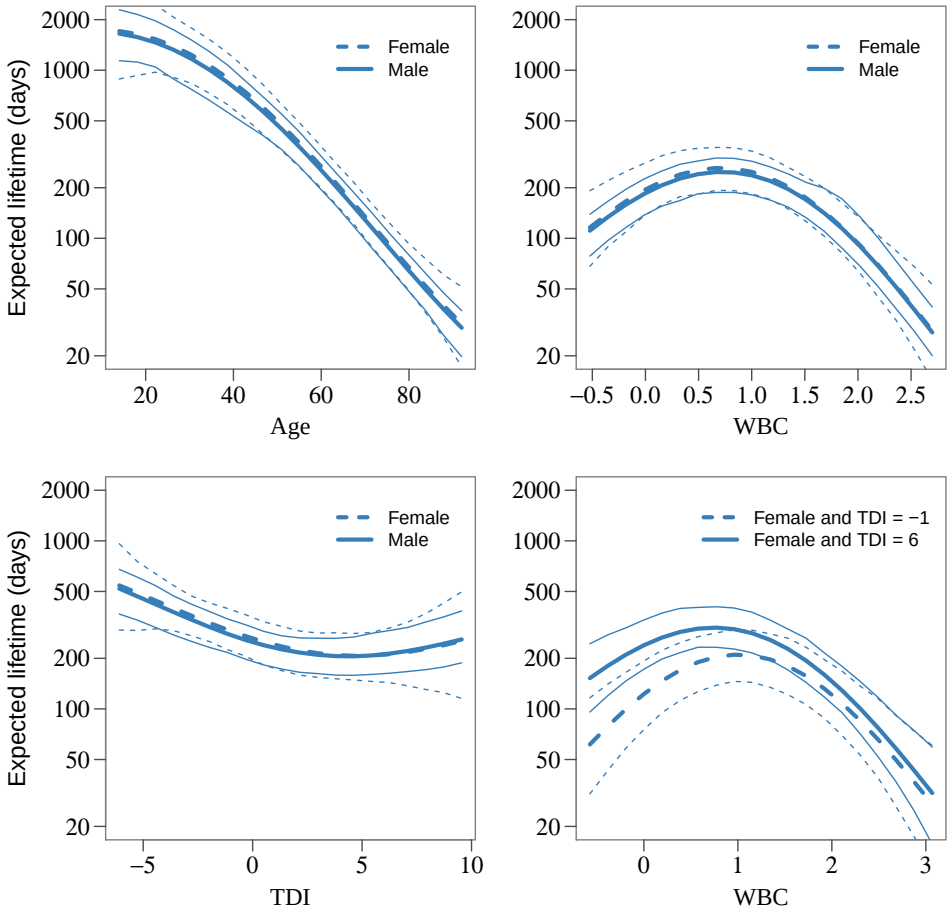

Fig. 17: Expected lifetime conditional comparison for each predictor with other predictors fixed to their mean values. The thick line in each graph is the posterior mean estimated using a HSGP model, and the thin lines represent pointwise 95% credible intervals.

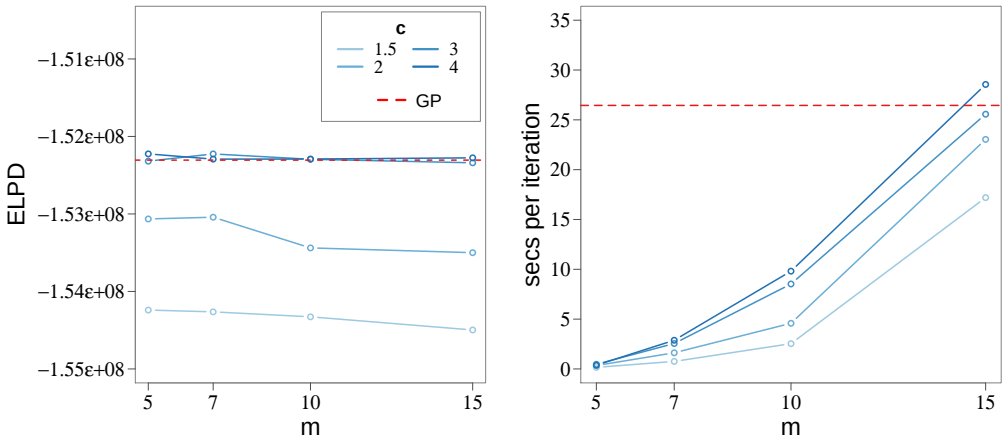

Fig. 18: Expected log predictive density (ELPD; left) and time of computation in seconds per iteration (iteration of the HMC sampling method; right) as a function of the number of basis functions m and boundary factor c.

The Stan model code for the exact GP and the approximate GP models of this case study can be found online at `https://github.com/gabriuma/basis_functions_approach_to_GP/tree/master/Paper/Case-study_Leukemia-data`.

6 Conclusion

Modeling unknown functions using exact GPs is computationally intractable for a lot of applications. This problem becomes especially severe when performing full Bayesian inference using sampling-based methods. In this paper, a recent approach for a low-rank representation of stationary GPs, originally proposed by Solin and Särkkä (2020), has been implemented and analyzed in detail. The method is based on a basis function approximation via Laplace eigenfunctions. The method has an attractive computational cost as it effectively approximates GPs by linear models, which is also an attractive property in modular probabilistic programming programming frameworks. The dominating cost per log density evaluation (during sampling) is $O(nm + m)$, which is a big benefit in comparison to $O(n^3)$ of a exact GP model. The obtained design matrix is independent of hyperparameters and therefore only needs to be constructed once, at cost $O(nm)$. All dependencies on the kernel and the hyperparameters are through the prior distribution of the regression weights. The parameters' posterior distribution is m-dimensional, where m is usually much smaller than the number of observations n.

The main contribution of this paper is an in-depth analysis and diagnosis of the performance and accuracy of the approximation in relation to the key factors of the method, that is, the number of basis functions, the boundary condition of the Laplace eigenfunctions, and the non-linearity of the function to be learned. Recommendations for the values of these key factors based on the recognized relations among them have been provided along with illustrations of these relations. These illustrations will not only help users to improve performance and save computation time, but also serve as a powerful diagnosis tool whether the chosen values for the number of basis functions and the boundary condition are adequate to fit to the data at hand with sufficient accuracy.

The developed approximate GPs can be easily applied as modular components in probabilistic programming frameworks such as Stan in both Gaussian and non-Gaussian observation models. Using several simulated and real datasets, we have demonstrated the practical applicability and improved sampling efficiency, as compared to exact GPs, of the developed method. The main drawback of the approach is that its computational complexity scales exponentially with the number of input dimensions. Hence, choosing optimal values for the number of basis functions and the boundary factor, using the recommendations and diagnostics provided in Figures **??** and **??**, is essential to avoid a excessive computational time especially in multivariate input spaces. However, in practice, input dimensionalities larger than three start to be quite computationally demanding even for moderately wiggly functions and few basis functions per input dimension. In these high dimensional cases, the proposed approximate GP methods may still be used for low-dimensional components in an additive modeling scheme but without modeling very high dimensional interactions, as complexity is linear with the number of additive components.

The obtained functional relationships between the key factors influencing the approximation not only help users to visually assess the accuracy of the method but can also serve an automatic diagnostic tool, if appropriately implemented. In this paper, we primarily studied the functional relationships for univariate inputs. Accordingly, investigating the functional relationships more thoroughly for multivariate inputs remains a topic for future research.

A Approximation of the covariance function using Hilbert space methods

In this section, we briefly present a summary of the mathematical details of the approximation of a stationary covariance function as a series expansion of eigenvalues and eigenfunctions of the Laplacian operator. This statement is based on the work by Solin and Särkkä (2020), who developed the mathematical theory behind the Hilbert Space approximation for stationary covariance functions.

Associated to each covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$ we can also define a covariance operator $\mathcal{K}$ over a function $f(\boldsymbol{x})$ as follows:

$$\mathcal{K}f(\boldsymbol{x}) = \int k(\boldsymbol{x}, \boldsymbol{x}')f(\boldsymbol{x}')\,\mathrm{d}\boldsymbol{x}'.$$

From the Bochner's and Wiener-Khintchine theorems, the spectral density of a stationary covariance function $k(\boldsymbol{x}, \boldsymbol{x}') = k(\boldsymbol{\tau}), \boldsymbol{\tau} = (\boldsymbol{x} - \boldsymbol{x}')$, is the Fourier transform of the covariance function,

$$S(\boldsymbol{w}) = \int k(\boldsymbol{\tau})e^{-2\pi i \boldsymbol{w}\boldsymbol{\tau}}\,\mathrm{d}\boldsymbol{\tau},$$

where $\boldsymbol{w}$ is in the frequency domain. The operator $\mathcal{K}$ will be translation invariant if the covariance function is stationary. This allows for a Fourier representation of the operator $\mathcal{K}$ as a transfer function which is the spectral density of the Gaussian process. Thus, the spectral density $S(\boldsymbol{w})$ also gives the approximate eigenvalues of the operator $\mathcal{K}$.

In the isotropic case $S(\boldsymbol{w}) = S(||\boldsymbol{w}||)$ and assuming that the spectral density function $S(\cdot)$ is regular enough, then it can be represented as a polynomial expansion:

$$S(||\boldsymbol{w}||) = a_0 + a_1||\boldsymbol{w}||^2 + a_2(||\boldsymbol{w}||^2)^2 + a_3(||\boldsymbol{w}||^2)^3 + \cdots . \quad \text{(A.1)}$$

The Fourier transform of the Laplace operator ∇^2 is $-||\boldsymbol{w}||$, thus the Fourier transform of $S(||\boldsymbol{w}||)$ is

$$\mathcal{K} = a_0 + a_1(-\nabla^2) + a_2(-\nabla^2)^2 + a_3(-\nabla^2)^3 + \cdots , \quad \text{(A.2)}$$

defining a pseudo-differential operator as a series of Laplace operators.

If the negative Laplace operator $-\nabla^2$ is defined as the covariance operator of the formal kernel l,

$$-\nabla^2 f(\boldsymbol{x}) = \int l(\boldsymbol{x}, \boldsymbol{x}') f(\boldsymbol{x}') \, \mathrm{d}\boldsymbol{x}',$$

then the formal kernel can be represented as

$$l(\boldsymbol{x}, \boldsymbol{x}') = \sum_j \lambda_j \phi_j(\boldsymbol{x}) \phi_j(\boldsymbol{x}'),$$

where $\{\lambda_j\}_{j=1}^{\infty}$ and $\{\phi_j(\boldsymbol{x})\}_{j=1}^{\infty}$ are the set of eigenvalues and eigenvectors, respectively, of the Laplacian operator. Namely, they satisfy the following eigenvalue problem in the compact subset $\boldsymbol{x} \in \Omega \subset \mathbb{R}^D$ and with the Dirichlet boundary condition (other boundary conditions could be used as well):

$$\begin{aligned} -\nabla^2 \phi_j(\boldsymbol{x}) &= \lambda \phi_j(\boldsymbol{x}), & x \in \Omega \\ \phi_j(\boldsymbol{x}) &= 0, & x \notin \Omega. \end{aligned}$$

Because $-\nabla^2$ is a positive definite Hermitian operator, the set of eigenfunctions $\phi_j(\cdot)$ are orthonormal with respect to the inner product

$$< f, g >= \int_\Omega f(\boldsymbol{x}) g(\boldsymbol{x}) \, \mathrm{d}(\boldsymbol{x})$$

that is,

$$\int_\Omega \phi_i(\boldsymbol{x}) \phi_j(\boldsymbol{x}) \, \mathrm{d}(\boldsymbol{x}) = \delta_{ij},$$

and all the eigenvalues λ_j are real and positive.

Due to normality of the basis of the representation of the formal kernel $l(\boldsymbol{x}, \boldsymbol{x}')$, its formal powers $s = 1, 2, \ldots$ can be written as

$$l(\boldsymbol{x}, \boldsymbol{x}')^s = \sum_j \lambda_j^s \phi_j(\boldsymbol{x}) \phi_j(\boldsymbol{x}'), \tag{A.3}$$

which are again to be interpreted to mean that

$$(-\nabla^2)^s f(\boldsymbol{x}) = \int l^s(\boldsymbol{x}, \boldsymbol{x}') f(\boldsymbol{x}') \, \mathrm{d}\boldsymbol{x}'.$$

This implies that we also have

$$[a_0 + a_1(-\nabla^2) + a_2(-\nabla^2)^2 + \cdots] f(\boldsymbol{x}) = \int [a_0 + a_1 l^1(\boldsymbol{x}, \boldsymbol{x}') + a_2 l^2(\boldsymbol{x}, \boldsymbol{x}') + \cdots] f(\boldsymbol{x}') \, \mathrm{d}\boldsymbol{x}'.$$

Then, looking at equations (A.2) and (A.3), it can be concluded

$$k(\boldsymbol{x}, \boldsymbol{x}') = \sum_j [a_0 + a_1 \lambda_j^1 + a_2 \lambda_j^2 + \cdots] \phi_j(\boldsymbol{x}) \phi_j(\boldsymbol{x}'). \tag{A.4}$$

By letting $||\boldsymbol{w}||^2 = \lambda_j$ the spectral density in eq. (A.1) becomes

$$S(\sqrt{\lambda_j}) = a_0 + a_1 \lambda_j + a_2 \lambda_j^2 + a_3 \lambda_j^3 + \cdots,$$

and substituting in eq. (A.4) then leads to the final form

$$k(\boldsymbol{x}, \boldsymbol{x}') = \sum_j S(\sqrt{\lambda_j}) \phi_j(\boldsymbol{x}) \phi_j(\boldsymbol{x}'), \tag{A.5}$$

where $S(\cdot)$ is the spectral density of the covariance function, λ_j is the jth eigenvalue and $\phi_j(\cdot)$ the eigenfunction of the Laplace operator in a given domain.

B Low-rank Gaussian process with a periodic covariance function

A GP model with a periodic covariance function does no fit in the framework of the HSGP approximation covered in this study, but it has also a low-rank representation. In this section, we first give a brief presentation of the results by Solin and Särkkä (2014), who obtain an approximate linear representation of a periodic squared exponential covariance function based on expanding the periodic covariance function into a series of stochastic resonators. Secondly, we analyze the accuracy of this approximation and, finally, we derive the GP model with this approximate periodic square exponential covariance function.

The periodic squared exponential covariance function takes the form

$$k(\tau) = \alpha \exp\left(-\frac{2 \sin^2(\omega_0 \frac{\tau}{2})}{\ell^2} \right), \tag{B.1}$$

where α is the magnitude scale of the covariance, ℓ is the characteristic length-scale of the covariance, and ω_0 is the angular frequency defining the periodicity.

Solin and Särkkä (2014) derive a cosine series expansion for the periodic covariance function (B.1) as follows,

$$k(\tau) = \alpha \sum_{j=0}^{J} \tilde{q}_j^2 \cos(j \omega_0 \tau), \tag{B.2}$$

which comes basically from a Taylor series representation of the periodic covariance function. The coefficients $\tilde{q}_j^2$

$$\tilde{q}_j^2 = \frac{2}{\exp(\frac{1}{\ell^2})} \sum_{j=0}^{\lfloor \frac{J-j}{2} \rfloor} \frac{(2\ell^2)^{-j-2}}{(j+i)! i!}, \tag{B.3}$$

where $j = 1, 2, \cdots, J$, and $\lfloor \cdot \rfloor$ denotes the floor round-off operator. For the index $j = 0$, the coefficient is

$$\tilde{q}_0^2 = \frac{1}{2} \frac{2}{\exp(\frac{1}{\ell^2})} \sum_{j=0}^{\lfloor \frac{J-j}{2} \rfloor} \frac{(2\ell^2)^{-j-2}}{(j+i)! i!}. \tag{B.4}$$

The covariance in eq. (B.2) is a Jth order truncation of a Taylor series representation. This approximation converges to eq. (B.1) when $J \to \infty$.

An upper bounded approximation to the coefficients $\tilde{q}_j^2$ and $\tilde{q}_0^2$ can be obtained by taking the limit $J \to \infty$ in the sub-sums in the corresponding equations (B.3) and (B.4), and thus leading to the following variance coefficients:

$$\begin{aligned} \tilde{q}_j^2 &= \frac{2 \mathrm{I}_j(\ell^{-2})}{\exp(\frac{1}{\ell^2})}, \\ \tilde{q}_0^2 &= \frac{\mathrm{I}_0(\ell^{-2})}{\exp(\frac{1}{\ell^2})}, \end{aligned} \tag{B.5}$$

for $j = 1, 2, \cdots, J$, and where the $\mathrm{I}_j(z)$ is the modified Bessel function (Abramowitz and Stegun, 1970) of the first kind. This approximation implies that the requirement of a valid covariance function is relaxed and only an optimal series approximation is required. A more detailed explanation and mathematical proofs of this approximation of a periodic covariance function are provided by Solin and Särkkä (2014).

In order to assess the accuracy of this representation as a function of the number of cosine terms J considered in the approximation, an empirical evaluation is carried out in a similar way than that in Section 4 of this work. Thus, Figure B.1 shows the minimum number of terms J required

to achieve a close approximation to the exact periodic squared exponential kernel as a function of the length-scale of the kernel. We have considered an approximation to be close enough in terms of satisfying eq. (16) with $\varepsilon = 0.005 \int k(\tau)\,\mathrm{d}\tau$ (0.5% of the total area under the curve of the exact covariance function k). Since this is a series expansion of sinusoidal functions, the approximation does not depend on any boundary condition.

The function values of a GP model with this low-rank representation of the periodic exponential covariance function can be easily derived. Considering the identity

$$\cos(j\omega_0(x-x')) = \cos(j\omega_0 x)\cos(j\omega_0 x') + \sin(j\omega_0 x)\sin(j\omega_0 x'),$$

the covariance $k(\tau)$ in eq. (B.2) can be written as

$$k(x,x') \approx \alpha \left(\sum_{j=0}^{J} \tilde{q}_j^2 \cos(j\omega_0 x)\cos(j\omega_0 x') + \sum_{j=1}^{J} \tilde{q}_j^2 \sin(j\omega_0 x)\sin(j\omega_0 x') \right). \tag{B.6}$$

With this approximation for the periodic squared exponential covariance function $k(x,x')$, the approximate GP model $f(x) \sim \mathcal{GP}\left(0, k(x,x')\right)$ equivalently leads to a linear representation of $f(\cdot)$ via

$$f(x) \approx \alpha^{1/2} \left(\sum_{j=0}^{J} \tilde{q}_j \cos(j\omega_0 x)\beta_j + \sum_{j=1}^{J} \tilde{q}_j \sin(j\omega_0 x)\beta_{J+1+j} \right), \tag{B.7}$$

where $\beta_j \sim \mathrm{Normal}(0,1)$, with $j = 1,\ldots,2J+1$. The cosine $\cos(j\omega_0 x)$ and sinus $\sin(j\omega_0 x)$ terms do not depend on the covariance hyperparameters ℓ. The only dependence on the hyperparameter ℓ is through the coefficients $\tilde{q}_j$, which are J-dimensional. The computational cost of this approximation scales as $O\big(n(2J+1)+(2J+1)\big)$, where n is the number of observations and J the number of cosine terms. The parameterization in eq. (B.7) is naturally in the non-centered form with independent prior distributions on β_j, which makes posterior inference easier.

Acknowledgements We thank Academy of Finland (grants 298742, 308640, and 313122), Finnish Center for Artificial Intelligence, and Technology Industries of Finland Centennial Foundation (grant 70007503; Artificial Intelligence for Research and Development) for partial support of this research. We also acknowledge the computational resources provided by the Aalto Science-IT project.

References

Abramowitz M, Stegun I (1970) Handbook of Mathematical Functions. Dover Publishing, New York

Adler RJ (1981) The Geometry of Random Fields. SIAM

Akhiezer NI, Glazman IM (1993) Theory of Linear Operators in Hilbert Space. Dover, New York

Andersen MR, Vehtari A, Winther O, Hansen LK (2017) Bayesian inference for spatio-temporal spike-and-slab priors. Journal of Machine Learning Research 18(139):1–58

Betancourt M (2017) A conceptual introduction to Hamiltonian Monte Carlo. arXiv preprint arXiv:170102434

Betancourt M, Girolami M (2019) Hamiltonian Monte Carlo for hierarchical models. In: Current Trends in Bayesian Methodology with Applications, Chapman and Hall/CRC, pp 79–101

Briol FX, Oates C, Girolami M, Osborne MA, Sejdinovic D (2015) Probabilistic integration: A role in statistical computation? arXiv preprint arXiv:151200933

Brooks S, Gelman A, Jones G, Meng XL (2011) Handbook of Markov Chain Monte Carlo. CRC Press

Bürkner PC (2017) brms: An R package for Bayesian multilevel models using Stan. Journal of Statistical Software 80(1):1–28

Burt D, Rasmussen CE, Van Der Wilk M (2019) Rates of convergence for sparse variational Gaussian process regression. In: Chaudhuri K, Salakhutdinov R (eds) Proceedings of the 36th International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, vol 97, pp 862–871

Carlin BP, Gelfand AE, Banerjee S (2014) Hierarchical Modeling and Analysis for Spatial Data. Chapman and Hall/CRC

Carpenter B, Gelman A, Hoffman MD, Lee D, Goodrich B, Betancourt M, Brubaker M, Guo J, Li P, Riddell A (2017) Stan: A probabilistic programming language. Journal of Statistical Software 76(1)

Carvalho CM, Polson NG, Scott JG (2010) The horseshoe estimator for sparse signals. Biometrika 97(2):465–480

Cramér H, Leadbetter MR (2013) Stationary and related stochastic processes: Sample function properties and their applications. Courier Corporation

Csató L, Fokoué E, Opper M, Schottky B, Winther O (2000) Efficient approaches to Gaussian process classification. In: Advances in neural information processing systems, pp 251–257

Deisenroth MP, Fox D, Rasmussen CE (2015) Gaussian processes for data-efficient learning in robotics and control. IEEE Transactions on Pattern Analysis and Machine Intelligence 37(2):408–423

Diggle PJ (2013) Statistical Analysis of Spatial and Spatio-temporal Point Patterns. Chapman and Hall/CRC

Furrer EM, Nychka DW (2007) A framework to understand the asymptotic properties of kriging and splines. Journal of the Korean Statistical Society 36(1):57–76

Gal Y, Turner R (2015) Improving the Gaussian process sparse spectrum approximation by representing uncertainty in frequency inputs. In: Bach F, Blei D (eds) Proceedings of the 32nd International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, vol 37, pp 655–664

Gelman A, Carlin JB, Stern HS, Dunson DB, Vehtari A, Rubin DB (2013) Bayesian Data Analysis. Chapman and Hall/CRC

Gelman A, Hill J, Vehtari A (2020) Regression and Other Stories. Cambridge University Press

Gibbs MN, MacKay DJ (2000) Variational Gaussian process classifiers. IEEE Transactions on Neural Networks 11(6):1458–1464

GPy (2012) GPy: A Gaussian process framework in Python. URL `http://github.com/SheffieldML/GPy`

Grenander U (1981) Abstract Inference. John Wiley & Sons

Hennig P, Osborne MA, Girolami M (2015) Probabilistic numerics and uncertainty in computations. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 471(2179):20150142

Hensman J, Durrande N, Solin A (2017) Variational fourier features for Gaussian processes. The Journal of Machine Learning Research 18(1):5537–5588

Jo S, Choi T, Park B, Lenk P (2019) bsamGP: An R package for Bayesian spectral analysis models using Gaussian process priors. Journal of Statistical Software, Articles 90(10):1–41

Lázaro Gredilla M (2010) Sparse Gaussian processes for large-scale machine learning. PhD thesis, Universidad Carlos III de Madrid

Loève M (1977) Probability Theory. Springer-Verlag, New York

Matthews AGdG, van der Wilk M, Nickson T, Fujii K, Boukouvalas A, León-Villagrá P, Ghahramani Z, Hensman J (2017) GPflow: A Gaussian process library using TensorFlow. Journal of Machine Learning Research 18(40):1–6

Minka TP (2001) Expectation propagation for approximate Bayesian inference. In: Proceedings of the Seventeenth Conference on Uncer-

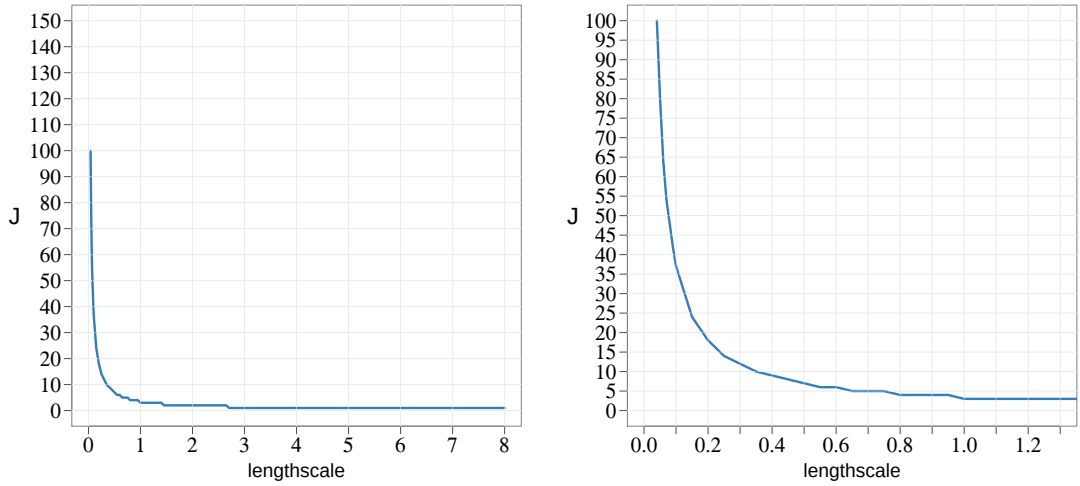

Fig. B.1: Relation among the minimum number of terms J in the approximation and the length-scale (ℓ) of the periodic squared exponential covariance function. The right-side plot is a zoom in of the left-side plot.

tainty in Artificial Intelligence, Morgan Kaufmann Publishers Inc., pp 362–369

Neal RM (1997) Monte Carlo implementation of Gaussian process models for Bayesian regression and classification. arXiv preprint physics/9701026

Piironen J, Vehtari A (2017a) On the hyperprior choice for the global shrinkage parameter in the horseshoe prior. In: Singh A, Zhu J (eds) Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, PMLR, Proceedings of Machine Learning Research, vol 54, pp 905–913

Piironen J, Vehtari A (2017b) Sparsity information and regularization in the horseshoe and other shrinkage priors. Electronic Journal of Statistics 11(2):5018–5051

Quiñonero-Candela J, Rasmussen CE (2005) A unifying view of sparse approximate Gaussian process regression. Journal of Machine Learning Research 6(Dec):1939–1959

Quiñonero-Candela J, Rasmussen CE, Figueiras-Vidal AR (2010) Sparse spectrum Gaussian process regression. Journal of Machine Learning Research 11(Jun):1865–1881

R Core Team (2019) R: A language and environment for statistical computing. Http://www.R-project.org/

Rahimi A, Recht B (2008) Random features for large-scale kernel machines. In: Platt JC, Koller D, Singer Y, Roweis ST (eds) Advances in Neural Information Processing Systems 20, Curran Associates, Inc., pp 1177–1184

Rahimi A, Recht B (2009) Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In: Koller D, Schuurmans D, Bengio Y, Bottou L (eds) Advances in Neural Information Processing Systems 21, Curran Associates, Inc., pp 1313–1320

Rasmussen CE, Nickisch H (2010) Gaussian processes for machine learning (GPML) toolbox. The Journal of Machine Learning Research 11:3011–3015

Rasmussen CE, Williams CK (2006) Gaussian Processes for Machine Learning. MIT press

Roberts SJ (2010) Bayesian Gaussian processes for sequential prediction, optimisation and quadrature. PhD thesis, University of Oxford

Solin A, Särkkä S (2014) Explicit link between periodic covariance functions and state space models. In: Kaski S, Corander J (eds) Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, PMLR, Proceedings of Machine Learning Research, vol 33, pp 904–912

Solin A, Särkkä S (2020) Hilbert space methods for reduced-rank Gaussian process regression. Statistics and Computing 30(2):419–446, much of the work in this paper is based on the pre-print version predating the published paper. Pre-print available at `https://arxiv.org/abs/1401.5508`.

Särkkä S, Solin A, Hartikainen J (2013) Spatiotemporal learning via infinite-dimensional Bayesian filtering and smoothing: A look at Gaussian process regression through Kalman filtering. IEEE Signal Processing Magazine 30(4):51–61

Van Trees HL (1968) Detection, Estimation, and Modulation Theory, Part I: Detection, Estimation, and Linear Modulation Theory. John Wiley & Sons, New York, NY

Vanhatalo J, Riihimäki J, Hartikainen J, Jylänki P, Tolvanen V, Vehtari A (2013) GPstuff: Bayesian modeling with Gaussian processes. The Journal of Machine Learning Research 14(1):1175–1179

Vehtari A, Ojanen J (2012) A survey of Bayesian predictive methods for model assessment, selection and comparison. Statistics Surveys 6:142–228

Vehtari A, Gelman A, Gabry J (2017) Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. Statistics and Computing 27(5):1413–1432

Wahba G (1990) Spline Models for Observational Data, vol 59. SIAM

Williams CK, Barber D (1998) Bayesian classification with Gaussian processes. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(12):1342–1351

Wood SN (2003) Thin plate regression splines. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 65(1):95–114

Wood SN (2011) Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 73(1):3–36