Spatial Statistics 28 (2018) 59–78

Contents lists available at ScienceDirect

Spatial Statistics

journal homepage: www.elsevier.com/locate/spasta

Spatial mapping with Gaussian processes and nonstationary Fourier features

Jean-Francois Ton [a], Seth Flaxman [b], Dino Sejdinovic [a], Samir Bhatt [c,*]

[a] Department of Statistics, University of Oxford, Oxford, OX1 3LB, UK
[b] Department of Mathematics and Data Science Institute, Imperial College London, London, SW7 2AZ, UK
[c] Department of Infectious Disease Epidemiology, Imperial College London, London, W2 1PG, UK

ARTICLE INFO

Article history:
Received 15 November 2017
Accepted 26 February 2018
Available online 29 March 2018

Keywords:
Gaussian process
Nonstationary
Spatial statistics
Random Fourier features

ABSTRACT

The use of covariance kernels is ubiquitous in the field of spatial statistics. Kernels allow data to be mapped into high-dimensional feature spaces and can thus extend simple linear additive methods to nonlinear methods with higher order interactions. However, until recently, there has been a strong reliance on a limited class of stationary kernels such as the Matérn or squared exponential, limiting the expressiveness of these modelling approaches. Recent machine learning research has focused on spectral representations to model arbitrary stationary kernels and introduced more general representations that include classes of nonstationary kernels. In this paper, we exploit the connections between Fourier feature representations, Gaussian processes and neural networks to generalise previous approaches and develop a simple and efficient framework to learn arbitrarily complex nonstationary kernel functions directly from the data, while taking care to avoid overfitting using state-of-the-art methods from deep learning. We highlight the very broad array of kernel classes that could be created within this framework. We apply this to a time series dataset and a remote sensing problem involving land surface temperature in Eastern Africa. We show that without increasing the computational or storage complexity, nonstationary kernels can be used to improve generalisation performance and provide more interpretable results.

© 2018 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

* Corresponding author.
E-mail address: bhattsamir@gmail.com (S. Bhatt).

https://doi.org/10.1016/j.spasta.2018.02.002
2211-6753/© 2018 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

The past decade has seen a tremendous and ubiquitous growth in both data collection and computational resources available for data analysis. In particular, spatiotemporal modelling has been brought to the forefront with applications in finance (Pace et al., 2000), weather forecasting (Berrocal et al., 2007), remote sensing (Wan et al., 2002; Weiss et al., 2014a, b, 2015) and demographic/disease mapping (Bhatt et al., 2013, 2015; Hay et al., 2013). The methodological workhorse of mapping efforts has been Gaussian process (GP) regression (Rasmussen and Williams, 2006; Diggle and Ribeiro, 2007). The reliance on GPs stems from their convenient mathematical framework which allows the modelling of distributions over nonlinear functions. GPs offer robustness to overfitting, a principled way to integrate over hyperparameters, and provide uncertainty intervals. In a GP, every point in some continuous input space is associated with a normally distributed random variable, such that every finite collection of those random variables has a multivariate normal distribution — entirely defined through a mean function $\mu(\cdot)$ and a covariance kernel function $k(\cdot, \cdot)$. In many settings, $\mu(\cdot) = 0$ and modelling proceeds through selecting the appropriate kernel function which entirely determines the properties of the GP, and can have a significant influence on both the predictive performance and on the model interpretability (Paciorek and Schervish, 2006; Genton et al., 2001). However, in practice, the kernel function is often (somewhat arbitrarily) set *a priori* to the squared exponential or Matérn class of kernels (Genton et al., 2001), justifying this choice by the fact that they model a rich class of functions (Micchelli et al., 2006).

While offering an elegant mathematical framework, performing inference with GP models is computationally demanding. Namely, evaluating the GP posterior involves a matrix inversion, which for n observations, requires $\mathcal{O}(n^3)$ time and $\mathcal{O}(n^2)$ storage complexity. This makes fitting full GP models prohibitive for any dataset that exceeds a few thousand observations (thereby limiting their use exactly in the regimes where a flexible nonlinear model is of interest). In response to these limitations, many scalable approximations to GP modelling have been proposed. Very broadly these can be catagorised into (a) low rank approximations, (b) sparse approximation methods and (c) spectral methods. Low rank approximations typically decompose the covariance matrix into a smaller rank m matrix to reduce the computational complexity to $\mathcal{O}(nm^2)$. Popular examples include inducing points representations (Quiñonero-Candela and Rasmussen, 2005), the Nyström approximation (Rasmussen and Williams, 2006), the fully independent training conditional (FITC) model (Snelson and Ghahramani, 2012), fixed rank Kriging (Cressie and Johannesson, 2008; Kang and Cressie, 2011), process convolutions (Higdon, 2002), empirical orthogonal decompositions (Obled and Creutin, 1986; Wikle and Cressie, 1999) and the multi-resolution approximation (Katzfuss, 2017). Sparse approximation methods, move away from using global basis functions as in the low rank approaches and focus on compactly supported covariance representations. Popular examples include covariance tapering (Furrer et al., 2006), solutions to stochastic partial differential equations (Lindgren et al., 2011), Gauss Markov random field approximations (Rue and Tjelmeland, 2002), and nearest neighbour Gaussian processes (Datta et al., 2016). Spectral methods appeal to spectral constructions of the covariance matrix, with popular examples including spectral and multiresolution representations (Wikle et al., 2001), random Fourier features (Rahimi and Recht, 2008a; Lázaro-Gredilla et al., 2010), generalised spectral kernels (Samo and Roberts, 2015; Remes et al., 2017).

In this contribution, we will focus on large-scale Fourier representations of GPs. The random Fourier feature (RFF) approach RFFs implement an extremely simple, yet efficient idea: instead of relying on the implicit feature map associated with the kernel RFFs create an explicit, low-dimensional random Fourier feature map, obtained by estimating an empirical characteristic function (as opposed to common empirical orthogonal decompositions (Obled and Creutin, 1986) solving the eigen value problem) from a given spectral density (Feuerverger and Mureika, 1977; Sriperumbudur and Szabo, 2015). The advantage of this explicit low-dimensional feature representation is that, unlike low rank matrix approximations, it approximates the entire kernel function not just the kernel matrix. Through numerical experiments, it has also been demonstrated that kernel algorithms constructed using the approximate kernel do not suffer from significant performance degradation (Rahimi and Recht, 2008a, b, 2009). To provide some intuition to the reader we perform a simple simulation experiment in Appendix. For a more thorough treatment of the theoretical properties of RFF kernels, finite-sample

performance, uniform convergence bounds, and kernel approximation quality the reader is directed here (Sriperumbudur and Szabo, 2015; Li and Honorio, 2017; Rahimi and Recht, 2009, 2008a, b; Avron et al., 2017).

Large-scale Fourier representations of GPs traditionally rely on strong assumptions of the stationarity (or shift-invariance) of kernel functions, which is made in the vast majority of applications (and is indeed satisfied by the most often used squared exponential and Matérn kernels). Stationarity in the spatio-temporal data means that the similarity between two responses in space and time does not depend on the location and time itself, but only on the difference (or lag) between them, i.e. kernel function can be written as $k(x_1, x_2) = \kappa(x_1 - x_2)$ for some function κ. Several recent works (Lázaro-Gredilla et al., 2010; Wilson and Adams, 2013; Yang et al., 2015) consider flexible families of kernels based on Fourier representations, thus avoiding the need to choose a specific kernel a priori and allowing the kernel to be learned from the data, but these approaches are restricted to the stationary case. In many applications, particularly when data is rich, relaxing the assumption of stationarity can greatly improve generalisation performance (Paciorek and Schervish, 2006). To address this, recent work in Samo and Roberts (2015) and Genton et al. (2001) note that a more general spectral characterisation exists that includes nonstationary kernels (Yaglom, 1987; Genton et al., 2001) and uses it to construct nonstationary kernel families. In this paper, we build on the work of Samo and Roberts (2015), Lázaro-Gredilla et al. (2010), Remes et al. (2017) and develop a simple and practicable framework for learning spatiotemporal nonstationary kernel functions directly from the data by exploiting the connections between Fourier feature representations, Gaussian processes and neural networks (Rasmussen and Williams, 2006). Specifically, we directly learn frequencies in nonstationary spectral kernel representations using an appropriate neural network architecture, and adopt techniques used for deep learning regularisation (Srivastava et al., 2014) to prevent overfitting. We demonstrate the utility of the proposed method for learning nonstationary kernel functions in a time series example and in spatial mapping of land surface temperature in East Africa.

2. Methods and theory

2.1. Gaussian process regression

Gaussian process regression (GPR) takes a training dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ where $y_i \in \mathbb{R}$ is real-valued response/output and $x_i \in \mathbb{R}^D$ is a D-dimensional input vector. The response y_i and the input x_i are connected via the observation model

$$y_i = f(x_i) + \epsilon_i, \quad \epsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma_n^2), \quad i = 1, \ldots, n. \tag{1}$$

GPR is a Bayesian non-parametric approach that imposes a prior distribution on functions f, namely a GP prior, such that any vector $\mathbf{f} = [f(x_1), \ldots, f(x_m)]$ of a finite number of evaluations of f follows a multivariate normal distribution $\mathbf{f} \sim \mathcal{N}(0, K_{\mathbf{xx}})$, where the covariance matrix $K_{\mathbf{xx}}$ is created as a Gram matrix based on the kernel function evaluations, $[K_{\mathbf{xx}}]_{ij} = k(x_i, x_j)$. Throughout this paper we will assume that the mean function of the GP prior is $\mu = 0$, however, all the approaches in this paper can be easily extended to include a mean function (Bhatt et al., 2017). In stationary settings, $k(x_i, x_j) = \kappa(x_i - x_j)$ for some function $\kappa(\delta)$. A popular choice is the automatic relevance determination (ARD) kernel (Rasmussen and Williams, 2006), given by $\kappa(\delta) = \tau^2 \exp(-\delta^\top \Lambda \delta)$ where $\tau^2 > 0$ and $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_D)$. Kernel k will typically have hyperparameters θ (e.g. $\theta = [\tau, \lambda_1, \ldots, \lambda_D]$ for the ARD kernel) and one can thus consider a Bayesian hierarchical model:

$$\begin{aligned} \theta &\sim \pi(\theta) \\ f|\theta &\sim GP(0, k_\theta) \\ y_i|f, x_i, \theta &\sim \mathcal{N}(f(x_i), \sigma_n^2), \quad i = 1, \ldots, n. \end{aligned} \tag{2}$$

The posterior predictive distribution is straightforward to obtain from the conditioning properties of multivariate normal distributions. For a new input x^*, we can find the posterior predictive

distribution of the associated response y^*

$$p(y^*|x^*, \mathcal{D}, \theta) = \mathcal{N}(y^*; \mu_\theta, \sigma_\theta^2) \tag{3}$$

$$\mu_\theta = k_{x^*\mathbf{x}}(K_{\mathbf{xx}} + \sigma_n^2 I_n)^{-1} y \tag{4}$$

$$\sigma_\theta^2 = \sigma_n^2 + k(x^*, x^*) - k_{x^*\mathbf{x}}(K_{\mathbf{xx}} + \sigma_n^2 I_n)^{-1} k_{\mathbf{x}x^*}, \tag{5}$$

where $k_{\mathbf{x}x^*} = [k(x_1, x^*), \ldots, k(x_n, x^*)]^\top$, $k_{x^*\mathbf{x}} = k_{\mathbf{x}x^*}^\top$ and it is understood that the dependence on θ is through the kernel $k = k_\theta$. The computational complexity in prediction stems from the matrix inversion $(K_{\mathbf{xx}} + \sigma_n^2 I_n)^{-1}$. The marginal likelihood (also called model evidence) of the vector of outputs $\mathbf{y} = [y_1, \ldots, y_n]$, is given by $p(\mathbf{y}|\theta) = \int p(\mathbf{y}|\mathbf{f}, \theta) p(\mathbf{f}|\theta) d\mathbf{f}$, is obtained by integrating out the GP evaluations $\mathbf{f}$ from the likelihood of the observation model. Maximising the marginal likelihood over hyperparameters allows for automatic regularisation and hence for selecting an appropriate model complexity. For a normal observation model in (1), the log marginal likelihood is available in closed form

$$\log p(\mathbf{y}|\theta) = -\frac{n}{2}\log(2\pi) - \frac{1}{2}|K_{\mathbf{xx}} + \sigma_n^2 I_n| - \frac{1}{2}\mathbf{y}^T \left(K_{\mathbf{xx}} + \sigma_n^2 I_n\right)^{-1} \mathbf{y}. \tag{6}$$

Computing the inverse and determinant in (6) are computationally demanding — moreover, they need to be computed for every hyperparameter value θ under consideration. To allow for computational tractability, we will use an approximation of $K_{\mathbf{xx}}$ based on Fourier features (see Sections 2.3 and 2.4).

Alternative representations can easily be used such as the primal/dual representations in a closely related frequentist method, kernel ridge regression (KRR) (Hastie et al., 2009). In contrast to KRR, optimising the marginal likelihood as above retains the same computational complexity while providing uncertainty bounds and automatic regularisation without having to tune a regularisation hyperparameter. However, the maximisation problem of (6) is non-convex thereby limiting the chance of finding a global optimum, but instead relying on reasonable local optima (Rasmussen and Williams, 2006).

2.2. Random Fourier feature mappings

The Wiener–Khintchine theorem states that the power spectrum and the autocorrelation function of a random process constitute a Fourier pair. Given this, random Fourier feature mappings and similar methodologies (Remes et al., 2017; Yang et al., 2015; Samo and Roberts, 2015; Lázaro-Gredilla et al., 2010; Rahimi and Recht, 2008a) appeal to Bochner's theorem to reformulate the kernel function in terms of its spectral density.

Theorem 1 (*Bochner's Theorem*)**.** *A stationary continuous kernel $k(x_i, x_j) = \kappa(x_i - x_j)$ on $\mathbb{R}^d$ is positive definite if and only if $\kappa(\delta)$ is the Fourier transform of a non-negative measure.*

Hence, for an appropriately scaled shift invariant complex kernel $\kappa(\delta)$, i.e. for $\kappa(0) = 1$, Bochner's Theorem ensures that its inverse Fourier Transform is a probability measure:

$$k(x_1, x_2) = \int_{\mathbb{R}^d} e^{i\omega^T(x_1 - x_2)} \mathbb{P}(d\omega). \tag{7}$$

Thus, Bochner's Theorem introduces the duality between stationary kernels and the spectral measures $\mathbb{P}(d\omega)$. Note that the scale parameter of the kernel, i.e. $\sigma_f^2 = \kappa(0)$ can be trivially added back into the kernel construction by rescaling. Table 1 shows some popular kernel functions and their respective spectral densities.

By taking the real part of Eq. (7) (since we are commonly interested only in real-valued kernels in the context of GP modelling) and performing standard Monte Carlo integration, we can derive a finite-dimensional, reduced rank approximation of the kernel function

$$k(x_1, x_2) = \int_{\mathbb{R}^D} e^{i\omega^T(x_1 - x_2)} \mathbb{P}(d\omega) \tag{8}$$

Table 1
Summary table of kernels and their spectral densities.

Kernel name	$k(\delta)$	$p(\omega)$
Squared exponential	$e^{-\frac{(\|\delta\|_2^2)}{2\sigma}}, \sigma > 0$	$(2\pi)^{-\frac{D}{2}}\sigma^D \exp(-\frac{\sigma^2\|\omega\|_2^2}{2})$
Laplacian	$\exp(-\sigma\|\delta\|_1), \sigma > 0$	$\left(\frac{2}{\pi}\right)^{\frac{2}{D}} \prod_{i=1}^{D} \frac{\sigma}{\sigma^2+\omega_i^2}$
Matérn	$\frac{2^{1-\lambda}}{\Gamma(\lambda)}\left(\frac{\sqrt{(2\lambda)}\|\delta\|_2}{\sigma}\right)^{\lambda} K_\lambda\left(\frac{\sqrt{(2\lambda)}\|\delta\|_2}{\sigma}\right)$ $\lambda > 0, \sigma > 0$	$\frac{2^{D+\lambda}\pi^{\frac{D}{2}}\Gamma(\lambda+D/2)\lambda^\lambda}{\Gamma(\lambda)\sigma^{2\lambda}}\left(\frac{2\lambda}{\sigma^2}+4\pi^2\|\omega\|_2^2\right)^{-(\lambda+D/2)}$

$$= \mathbb{E}_{\omega\sim\mathbb{P}}\left[e^{i\omega^T(x_1-x_2)}\right], \tag{9}$$

$$= \mathbb{E}_{\omega\sim\mathbb{P}}\left[\cos(\omega^T(x_1-x_2)) + i\sin(\omega^T(x_1-x_2))\right] \tag{10}$$

$$= \mathbb{E}_{\omega\sim\mathbb{P}}\left[\cos(\omega^T(x_1-x_2))\right] \tag{11}$$

$$= \mathbb{E}_{\omega\sim\mathbb{P}}\left[\cos(\omega^T x_1)\cos(\omega^T x_2) + \sin(\omega^T x_1)\sin(\omega^T x_2)\right] \tag{12}$$

$$\approx \frac{1}{m}\sum_{k=1}^{m}\left(\cos(\omega_k^T x_1)\cos(\omega_k^T x_2) + \sin(\omega_k^T x_1)\sin(\omega_k^T x_2)\right) \tag{13}$$

$$= \frac{1}{m}\sum_{k=1}^{m}\Phi_k(x_1)^T\Phi_k(x_2) \tag{14}$$

where $\{\omega_k\}_{k=1}^m \overset{i.i.d.}{\sim} \mathbb{P}$ and we denoted

$$\Phi_k(x_l) = \begin{pmatrix}\cos(\omega_k^T x_l)\\ \sin(\omega_k^T x_l)\end{pmatrix}.$$

For a covariate design matrix $\mathbf{X} \in \mathbb{R}^{n\times D}$ (with rows corresponding to data vectors $x_1, \ldots, x_n$), and frequency matrix $\Omega \in \mathbb{R}^{m\times D}$ (with rows corresponding to frequencies $\omega_1, \ldots, \omega_m$), we let $\mathbf{\Phi_x} = \left[\cos(\mathbf{X}\Omega^\top)\ \sin(\mathbf{X}\Omega^\top)\right]$ be a $n \times 2$m matrix referred to as the feature map of the dataset. The estimated covariance matrix can be computed as $\widehat{K_{\mathbf{xx}}} = \frac{1}{m}\mathbf{\Phi_x}\mathbf{\Phi_x}^T$ which has rank at most 2m. Substituting $\widehat{K_{\mathbf{xx}}}$ into (6) now allows rewriting the determinant and the inverse in terms of the $2m\times 2$m matrix $\mathbf{\Phi_x}^T\mathbf{\Phi_x}$, thereby reducing the computational cost of inference from $\mathcal{O}(n^3)$ to $\mathcal{O}(nm^2)$, where m is the number of Monte Carlo samples/frequencies. Typically, $m \ll n$.

In particular, by defining $A = \mathbf{\Phi_x}^T\mathbf{\Phi_x} + m\frac{\sigma_n^2}{\sigma_f^2}I_{2m}$ where σ_n^2 is the observation noise variance and $\sigma_f^2 = \kappa(0)$ is the kernel scale parameter, and taking $R = \text{chol}(A)$ to be the Cholesky factor of A, we first calculate vectors α_1, α_2 solving the linear systems of equations $R\alpha_1 = \mathbf{\Phi_x}^T\mathbf{y}$, $R^T\alpha_2 = \alpha_1$ and $R^T\alpha_3 = \phi_{\mathbf{x}^*}$. The log marginal likelihood can then be computed efficiently as:

$$\log p(\mathbf{y}|\theta) = -\frac{1}{2\sigma_n^2}\left(\|\mathbf{y}\|^2 - \|\alpha_1\|^2\right) - \frac{1}{2}\sum_i \log(R_{ii}^2) + m\log\left(m\frac{\sigma_n^2}{\sigma_f^2}\right) - \frac{n}{2}\log(2\pi\sigma_n^2). \tag{15}$$

Additionally, the posterior predictive mean and variance can be estimated as

$$\widehat{\mu_\theta} = \frac{\sigma_f^2}{m}\mathbf{\Phi_{x^*}}^T\alpha_2 \tag{16}$$

$$\widehat{\sigma_\theta^2} = \sigma_n^2\left(1 + \frac{\sigma_f^2}{m}\|\alpha_3\|^2\right). \tag{17}$$

There are two important disadvantages of standard random Fourier features as proposed by Rahimi and Recht (2008a): firstly, only stationary (shift invariant) kernels can be approximated, and secondly we have to select a priori a specific class of kernels and their corresponding spectral distributions

(e.g. Table 1). In this paper, we address both of these limitations, with a goal to construct methods to learn a nonstationary kernel from the data, while preserving the computational efficiency of random Fourier features.

While we can think about the quantities in (16) as giving approximations to the full GP inference with a given kernel k, they are in fact performing exact GP calculations for another kernel $\hat{k}$ defined using the explicit feature map $\mathbf{\Phi}_{\boldsymbol{x}}$ defined through frequencies sampled from the spectral measure of k. We can thus think about these feature maps as parametrising a family of kernels in their own right and treat frequencies $\omega_1, \ldots, \omega_m$ as kernel parameters to be optimised, i.e. learned from the data by maximising the log marginal likelihood. It should be noted that dropping the imaginary part of our kernel symmetrises the spectral measure allowing us to use any $\mathbb{P}(d\omega)$ — regardless of its symmetry properties, we will still have a real-valued kernel. In particular, one can use an empirical spectral measure defined by any finite set of frequencies.

2.3. Nonstationary random Fourier features

Contrary to stationary kernels, which only depend on the lag vector i.e. $\delta = x_i - x_j$, nonstationary kernels depend on the inputs themselves. A simple example of a nonstationary kernel would be the polynomial kernel defined as:

$$k(x_1, x_2) = (x_1^\top x_2 + 1)^r. \tag{18}$$

To extend the stationary random feature mapping to nonstationary kernels, following Samo and Roberts (2015) Genton et al. (2001) and Yaglom (1987), we will need to use a more general spectral characterisation of positive definite functions which encompasses stationary *and* nonstationary kernels.

Theorem 2 (*Yaglom, 1987; Genton et al., 2001*)**.** *A nonstationary kernel $k(x_1, x_2)$ is positive definite in $\mathbb{R}^d$ if and only if it has the form:*

$$k(x_1, x_2) = \int_{\mathbb{R}^D \times \mathbb{R}^D} e^{i(w_1^T x_1 - w_2^T x_2)} \mu(dw_1, dw_2) \tag{19}$$

where $\mu(dw_1, dw_2)$ is the Lebesgue–Stieltjes measure associated to some positive semi-definite function $f(w_1, w_2)$ with bounded variation.

From the above theorem, a nonstationary kernel can be characterised by a spectral measure $\mu(d\omega_1, d\omega_2)$ on the product space $\mathbb{R}^D \times \mathbb{R}^D$. Again, without loss of generality we can assume that μ is a probability measure. If μ is concentrated along the diagonal, $\omega_1 = \omega_2$, we recover the spectral representation of stationary kernels in the previous section. However, exploiting this more general characterisation, we can construct feature mappings for nonstationary kernels.

Just like in the stationary case, we can approximate (19) using Monte Carlo integration. In order to ensure a valid positive semi-definite spectral density we first have to symmetrise $f(\omega_1, \omega_2)$ by ensuring $f(\omega_1, \omega_2) = f(\omega_2, \omega_1)$ and including the diagonal components $f(\omega_1, \omega_1)$ and $f(\omega_2, \omega_2)$ (Remes et al., 2017). We can take a general form of density g on the product space and "symmetrise":

$$f(\omega_1, \omega_2) = \frac{1}{4}\left(g(\omega_1, \omega_2) + g(\omega_2, \omega_1) + g(\omega_1, \omega_1) + g(\omega_2, \omega_2)\right).$$

Once again using Monte Carlo integration we get:

$$\begin{aligned} k(x_1, x_2) &= \frac{1}{4}\int_{\mathbb{R}^D \times \mathbb{R}^D} \left(e^{i(\omega_1^T x_1 - \omega_2^T x_2)} + e^{i(\omega_2^T x_1 - \omega_1^T x_2)} + e^{i(\omega_1^T x_1 - \omega_1^T x_2)} + e^{i(\omega_2^T x_1 - \omega_2^T x_2)}\right) \mu(d\omega_1 d\omega_2) \\ &= \frac{1}{4}\mathbb{E}_\mu\left(e^{i(\omega_1^T x_1 - \omega_2^T x_2)} + e^{i(\omega_2^T x_1 - \omega_1^T x_2)} + e^{i(\omega_1^T x_1 - \omega_1^T x_2)} + e^{i(\omega_2^T x_1 - \omega_2^T x_2)}\right) \\ &\approx \frac{1}{4m}\sum_{k=1}^{m}\left(e^{i(x_1^T \omega_k^1 - x_2^T \omega_k^2)} + e^{i(x_1^T \omega_k^2 - x_2^T \omega_k^1)} + e^{i(x_1^T \omega_k^1 - x_2^T \omega_k^1)} + e^{i(x_1^T \omega_k^2 - x_2^T \omega_k^2)}\right) \end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{4m}\sum_{k=1}^{m}\Big\{\cos(x_1^T\omega_k^1)\cos(x_2^T\omega_k^1)+\cos(x_1^T\omega_k^1)\cos(x_2^T\omega_k^2)\\
&\quad+\cos(x_1^T\omega_k^2)\cos(x_2^T\omega_k^1)+\cos(x_1^T\omega_k^2)\cos(x_2^T\omega_k^2)\\
&\quad+\sin(x_1^T\omega_k^1)\sin(x_2^T\omega_k^1)+\sin(x_1^T\omega_k^1)\sin(x_2^T\omega_k^2)\\
&\quad+\sin(x_1^T\omega_k^2)\sin(x_2^T\omega_k^1)+\sin(x_1^T\omega_k^2)\sin(x_2^T\omega_k^2)\Big\}\quad (\textit{taking the real part})\\
&= \frac{1}{4m}\sum_{k=1}^{m}\Phi_k(x_1)^T\Phi_k(x_2)
\end{aligned}
$$

where $\{(\omega_k^1,\omega_k^2)\}_{k=1}^m \overset{i.i.d.}{\sim} \mu$ and

$$\Phi_k(x_l)=\begin{pmatrix}\cos(x_l^T\omega_k^1)+\cos(x_l^T\omega_k^2)\\ \sin(x_l^T\omega_k^1)+\sin(x_l^T\omega_k^2)\end{pmatrix}.$$

Hence, by denoting $\Omega^l\in\mathbb{R}^{m\times D}$ (with rows corresponding to frequencies $\omega_1^l,\ldots,\omega_m^l$) for $l=1,2$ as before, we obtain the corresponding feature map for the approximated kernel as an $n\times 2m$ matrix

$$x\to\Phi_{\boldsymbol{x}}=\left[\cos(\mathbf{X}(\Omega^1)^T)+\cos(\mathbf{X}(\Omega^2)^T)\mid\sin(\mathbf{X}(\Omega^1)^T)+\sin(\mathbf{X}(\Omega^2)^T)\right] \tag{20}$$

and can be condensed to an identical form as in the stationary case.

$$\widehat{K_{\mathbf{xx}}}=\frac{1}{4m}\Phi_{\boldsymbol{x}}\Phi_{\boldsymbol{x}}{}^T. \tag{21}$$

The non-stationarity in Eq. (21) arises from the product of differing locations x_1 and x_2 by different frequencies ω_k^1,ω_k^2, hence making the kernel dependent on the values of x_1 and x_2 and not only the lag vector. If the frequencies were exactly the same we just revert to the stationary case. The complete construction of random Fourier feature approximation is summarised in the algorithm below.

Algorithm 1 Random Fourier features for nonstationary kernels

Input: spectral measure μ, dataset $\mathbf{X}$, number of frequencies m
Output: Approximation to $K_{\mathbf{xx}}$
Start Algorithm:
Sample m pairs of frequencies $\{(\omega_k^1,\omega_k^2)\}_{k=1}^m \overset{i.i.d.}{\sim}\mu$ giving Ω^1 and Ω^2
Compute $\Phi_{\boldsymbol{x}}=\left[\cos(\mathbf{X}(\Omega^1)^T)+\cos(\mathbf{X}(\Omega^2)^T)\mid\sin(\mathbf{X}(\Omega^1)^T)+\sin(\mathbf{X}(\Omega^2)^T)\right]\in\mathbb{R}^{n\times 2m}$
$\widehat{K_{\mathbf{xx}}}=\frac{1}{4m}\Phi_{\boldsymbol{x}}\Phi_{\boldsymbol{x}}{}^T$
End Algorithm

However, just like in the stationary case, we can think about nonstationary Fourier feature maps as parametrising a family of kernels and treat frequencies $\{(\omega_k^1,\omega_k^2)\}_{k=1}^m$ as kernel parameters to be learned by maximising the log marginal likelihood, which is an approach we pursue in this work. Again, symmetrisation due to dropping imaginary parts implies that any empirical spectral measure is valid (there are no constraints on the frequencies).

2.4. On the choice of spectral measure in non-stationary case

Using the characterisation in Eq. (21) one only requires the specification of the (Lebesgue–Stieltjes measurable) distribution $f(\omega_1,\omega_2)$ in order to construct a nonstationary kernel. This very general formulation allows us to create the full spectrum encompassing both simple and highly complex kernels.

In the simplest case, $f(\omega_1,\omega_2)=f_1(\omega_1)f_2(\omega_2)$, i.e. it can be a product of popular spectral densities listed in Table 1. Furthermore, one could consider cases where these individual spectral densities

factorise further across dimensions. This corresponds to a notion of *separability*. In spatio-temporal data, separability can be very useful as it enables interpretation of the relationship between the covariates as well as computationally efficient estimations and inferences (Finkenstadt et al., 2007). Practical implementation is straightforward; consider the classic spatio-temporal setting with 3 covariates — longitude, latitude and time. When drawing random samples of $\omega_l = (\omega_l^1, \omega_l^2, \omega_l^3)$ where $l \in \{1, 2\}$, we could define the ω_l^i to come from different distributions, allowing us to individually model each input dimension. If the distribution on frequencies is independent across dimensions then we see that if $\omega_1 = (\omega_1^1, \omega_1^2, \omega_1^3)$ and $\omega_2 = (\omega_2^1, \omega_2^2, \omega_2^3)$:

$$k(x_1, x_2) = \int e^{i\omega_1^T x_1 - i\omega_2^T x_2} f(\omega_1, \omega_2) d\omega_1 d\omega_2 \tag{22}$$

$$= \int e^{i(\omega_1^1 x_1^1 + \omega_1^2 x_1^2 + \omega_1^3 x_1^3 - \omega_2^1 x_2^1 - \omega_2^2 x_2^2 - \omega_2^3 x_2^3)} \prod_{p=1}^{3} f(\omega_1^p, \omega_2^p) d\omega_1^p d\omega_2^p \tag{23}$$

$$= k_1(x_1^1, x_2^1) k_2(x_1^2, x_2^2) k_3(x_1^3, x_2^3). \tag{24}$$

A practical example for spatio-temporal modelling of a nonstationary separable kernel would be generating a four dimensional $(\omega_1^1, \omega_2^1, \omega_1^2, \omega_2^2)$, sample from independent Gaussian distributions (whose spectral density corresponds to a squared exponential kernel) representing nonstationary spatial coordinates, and a two dimensional (ω_1^3, ω_2^3) from a Student-t distribution with 0.5 degrees of freedom (whose spectral density corresponds to a Matérn 1/2 kernel or exponential kernel) representing temporal coordinates.

To move from separable to non-separable, nonstationary kernels one only needs to introduce some dependence structure within ω^1 or ω^2 i.e. across feature dimensions, such as for example using the multivariate normal distribution in $\mathbb{R}^D$, in order to prevent the factorisation in Eq. (22). The correlation structure in these multivariate distributions is what creates the non-separability.

To create non-separable kernels with different spectral densities along each feature dimension copulas can be used. An example in a spatial (latitude, longitude feature dimensions) setting using the Gaussian copula, would involve generating samples for ω^1 or $\omega^2 \in \mathbb{R}^2$ (or both) from a multivariate normal distribution $\{\omega_k^1\}_{k=1}^m \overset{i.i.d.}{\sim} \mathcal{N}(0, \Sigma)$, pass these through the Gaussian cumulative distribution function, and then passed through the quantile function of another distribution (Λ) i.e. $\mathcal{C}_\Lambda(\omega^1) = CDF_\Lambda(CDF_\mathcal{N}^{-1}(\omega^1))$. This transformation can also be done using different Λs along different feature dimensions. Alternative copulas can be readily used, including the popular Archimedian Copulas: Clayton, Frank and Gumbel (Genest and MacKay, 1986). Additionally, mixtures of multivariate normals can be used (Remes et al., 2017; Yang et al., 2015) to create arbitrarily complex non-separable and nonstationary kernels. Given sufficient components any probability density function can be approximated to the desired accuracy.

In this paper, we focus on the most general case where the frequencies $\{(\omega_k^1, \omega_k^2)\}_{k=1}^m$ are treated as kernel parameters and are learnt directly from the data by optimising the marginal likelihood, i.e. they are not associated to any specific family of joint distributions. This approach allows us to directly learn nonstationary kernels of arbitrary complexity as m increases. However, a major problem with such a heavily overparametrized kernel is the possibility of overfitting. Stationary examples of learning frequencies directly from the data (Lázaro-Gredilla et al., 2010; Gal and Turner, 2015; Tan et al., 2013) have been known to overfit despite the regularisation due to working with marginal likelihood. This problem is further exacerbated in high-dimensional settings, such as those in spatio-temporal mapping with covariates. In this paper, we include an additional regularisation inspired by dropout (Srivastava et al., 2014) which prevents the co-adaptation of the learnt frequencies ω_1, ω_2.

2.5. Gaussian dropout regularisation

Dropout (Srivastava et al., 2014) is a regularisation technique introduced to mitigate overfitting in deep neural networks. In its simplest form, dropout involves setting features/matrix entries to zero with probability $q = 1-p$, i.e. according to a *Bernoulli*(p) for each feature. The main motivation behind

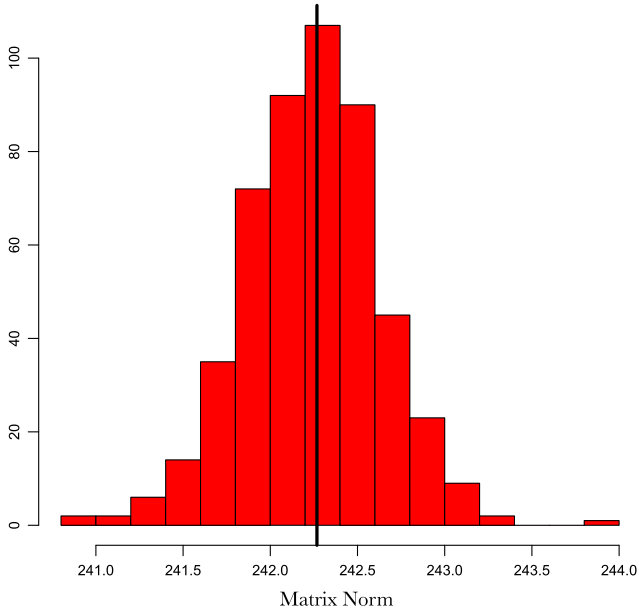


Fig. 1. Histogram of the Euclidean norm of a covariance matrix $\Phi\Phi^T$ with Gaussian dropout of $\sigma_p = 0.05$. The black line is the norm of $\Phi\Phi^T$ *without* noise.

the algorithm is to prevent co-adaptation by forcing features to be robust and rely on population behaviour. This prevents individual features from overfitting to idiosyncrasies of the data.

Using standard dropout, where zeros are introduced into the frequencies $\{(\omega_k^1, \omega_k^2)\}_{k=1}^m$ can be problematic due to the trigonometric transformations in the projected features. An alternative to dropout that has been shown to be just as effective if not better is Gaussian dropout (Srivastava et al., 2014; Baldi and Sadowski, 2013). Regularisation via Gaussian dropout involves augmenting our sample distribution as $\{(\omega_k^1, \omega_k^2)_\eta\}_{k=1}^m = \mathcal{N}(1, \sigma_p^2) \odot \{(\omega_k^1, \omega_k^2)\}_{k=1}^m$. The addition of noise through $\mathcal{N}(1, \sigma_p^2)$ ensures unbiased estimates of the covariance matrix i.e. $\mathbb{E}[\{(\omega_k^1, \omega_k^2)_\eta\}_{k=1}^m] = \mathbb{E}[\{(\omega_k^1, \omega_k^2)\}_{k=1}^m]$ (see Fig. 1). As with dropout, this approach prevented our population Monte Carlo sample from co-adapting, and ensured that the learnt frequencies are robust and not overfitting noise in the data. An additional benefit of this procedure over improving generalisation error and preventing overfitting was to speed up the convergence of gradient descent optimisers through escaping saddle points more effectively (Lee et al., 2017). The noise parameter σ_p defines the degree of regularisation and is a hyperparameter that needs to be tuned. However, we found in practice when coupled with an early stopping procedure, learning the frequencies is robust to sensible choices of σ_p.

3. Implementation details

All of the modelling was performed within the TensorFlow framework (Abadi et al., 2016). Optimisation was performed using ADAM (Kingma and Ba, 2014) gradient descent. In addition to Gaussian dropout, additional regularisation was introduced by using early stopping (Prechelt, 1998). The early stopping criteria used followed the recommendations in Prechelt (1998), where the relative cross validation test accuracy was monitored and early stopping of the optimisation process was performed if the testing accuracy was not improved after a certain number of gradient decent steps (termed the patience parameter Prechelt, 1998). Cross-validation was performed using a 70–30 split averaged over 20 independent runs and testing performance evaluated via the mean squared error and correlation. The testing performance of the posterior predictive distribution and uncertainty was

performed using the continuous ranked probability score (CRPS) (Matheson and Winkler, 1976) and the probability integral transform (PIT) (Held et al., 2010). A well calibrated model should have PIT scores distributed approximately $\sim$ *Uniform*(0, 1), and CRPS scores clustered towards zero.

The two hyperparameters not integrated out in the marginal likelihood: the dropout noise σ_p and the gradient descent learning rate were chosen via a random search (Bergstra and Bengio, 2012) over 500 samples, and parameters selected as those returning the lowest average mean squared cross validation testing error.

To choose the number of features we follow the theoretical recommendations of Rudi and Rosasco (2016) who show that $\mathcal{O}(1/\sqrt{(n)})$ generalisation learning bounds can be achieved under certain conditions with $\mathcal{O}(log(n)\sqrt{(n)})$ features (Rudi and Rosasco, 2016). These bounds result in the same prediction accuracy of the exact kernel ridge regression estimator. Starting from $m = log(n)\sqrt{(n)}$ features the final number of Fourier features was estimated by successively increasing m and monitoring the cross validation test accuracy. The optimal number of features was selected as those which asymptote the mean squared cross validation testing error. Code from this paper can be found at (*github.com/bhattsamir/Non_stationary$_f$eatures*).

4. Results

4.1. Google daily high stock price

To demonstrate the use of the developed method and the utility of nonstationary modelling, we consider time series data of the daily high stock price of Google spanning 3295 days from 19th August 2004 to 20th September 2017. We set $x \in \{1, \ldots, 3295\}$ and $y = \log(Stock_{high})$. For the stationary case we use vanilla random Fourier features (Rahimi and Recht, 2008a; Lázaro-Gredilla et al., 2010) with the squared exponential kernel (Gaussian spectral density) and $m = 600$ *fixed* frequencies. For the nonstationary case we use $m = 300$ frequencies for each ω_1 and for ω_2. We performed a sensitivity analysis to check that no improvements in either the log marginal likelihood or testing error resulted from using more features.

Fig. 2 (top left) shows the comparison in the optimisation paths of the negative log marginal likelihood between the two methods. It is clear that the nonstationary approach reaches a lower minima than the vanilla random Fourier features approach. This is also mirrored in the testing accuracy over the 20 independent runs where our approach achieves a mean squared error and correlation of 3.29×10^{-5} and 0.999, while the vanilla Fourier features approach achieves a mean squared error and correlation of 5.69×10^{-5} and 0.987. Of note is the impact of our Gaussian dropout regularisation, which, through the injection of noise, appears to converge faster and avoid plateaus. This is entirely in keeping with previous experiences using dropout variants (Baldi and Sadowski, 2013) and highlights an added benefit over using only ridge (weight decay) regularisation. Posterior predictive checks using the PIT and CRPS indicate good fits (see Fig. 4).

Fig. 3 (top) shows the overall fits compared to the raw data. Both methods appear to fit the data very well, as reflected in the testing statistics, but when examining a zoomed-in transect it is clear that the learnt nonstationary features fit the data better than the vanilla random features by allowing variable degree of smoothness in time. The combination of nonstationarity and kernel flexibility allowed us to learn a much better characterisation of the data patterns without overfitting. The covariance matrix comparisons (Fig. 3 bottom) further highlight this point where the learnt nonstationary covariance matrix shares some similarities with the vanilla random features covariance matrix, such as the concentration on the diagonal, but exhibits a much greater degree of texture. The histograms in Fig. 3 provide another perspective on the covariance structure, where the vanilla features are by design Gaussian distributed, but learnt nonstationary frequencies are far from Gaussian (Kolmogorov–Smirnov test p-value$< 10^{-16}$) exhibiting skewness and heavy tails. Additionally, the differences between the learnt frequencies ω_1 and ω_2 show that, not only is the learnt kernel far from Gaussian, but that it is indeed also nonstationary. This simple example also highlights the potential deficiencies of choosing kernels/frequencies *apriori*.

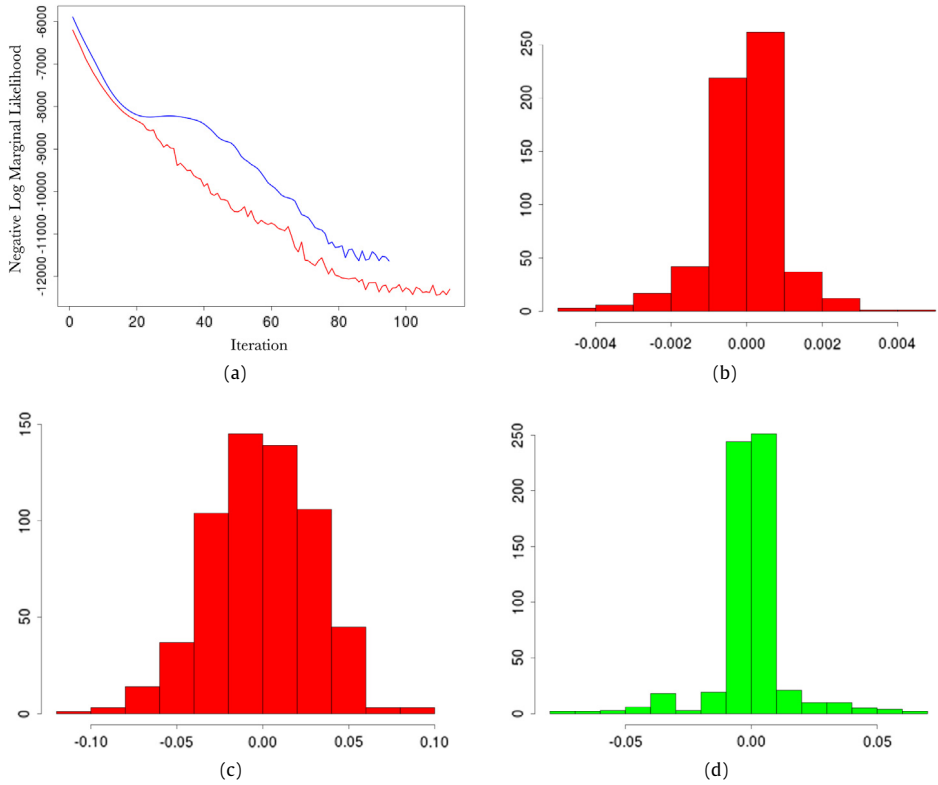


Fig. 2. (top left) Log marginal likelihood (Y-axis), optimisation gradient update count (X-axis), vanilla random features (blue), proposed approach (red); (top right) Histogram of learnt ω_1 for our nonstationary approach; (bottom left) Histogram of learnt ω_2 for our nonstationary approach; (bottom right) Histogram of ω_2 for vanilla random Fourier features. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4.2. Spatial temperature anomaly for East Africa in 2016

We next consider MOD11A2 Land Surface Temperature (LST) 8-day composite data (Wan et al., 2002), gap filled for cloud cover images (Weiss et al., 2014a) and averaged to a synoptic yearly mean for 2016. To replicate common situations for spatial mapping, such as interpolation from sparse remote sensed sites or cluster house hold survey locations (Bhatt et al., 2017) we randomly sample 6000 LST locations (only $\sim$ 5% of the total) from the East Africa region (see Figs. 6 and 7). We set $x \in \mathbb{R}^2 = \{Latitude, Longitude\}$ i.e. using only the spatial coordinates as covariates, and use the LST temperature anomaly as the response. We apply our nonstationary approach, learning a total of $m = 1500$ frequencies (750 sine and 750 cos). Cross validation was evaluated over all pixels excluding the training set ($\sim 83,000$) and averaged over 20 independent runs with testing performance evaluated via the mean squared error, correlation, CRPS and PIT. We also compare our fit to 3 of the best performing spatial statistics methods (Heaton et al., 2017) using the exact same data and using approximately the same number of basis functions. The three comparison methods were the SPDE approach (Lindgren et al., 2011), LatticeKrig (Nychka et al., 2015) and the multi-resolution approximation (MRA) (Katzfuss, 2017). Following Heaton et al. (2017) all three models were fit with a Matérn covariance function with $\nu = 1$. For the SPDE model 1539 mesh basis functions were chosen, for LatticeKrig 1534 basis functions were used (5 levels with weight 4.4 and NC = 2.15) and for MRA 1550 basis functions were used.

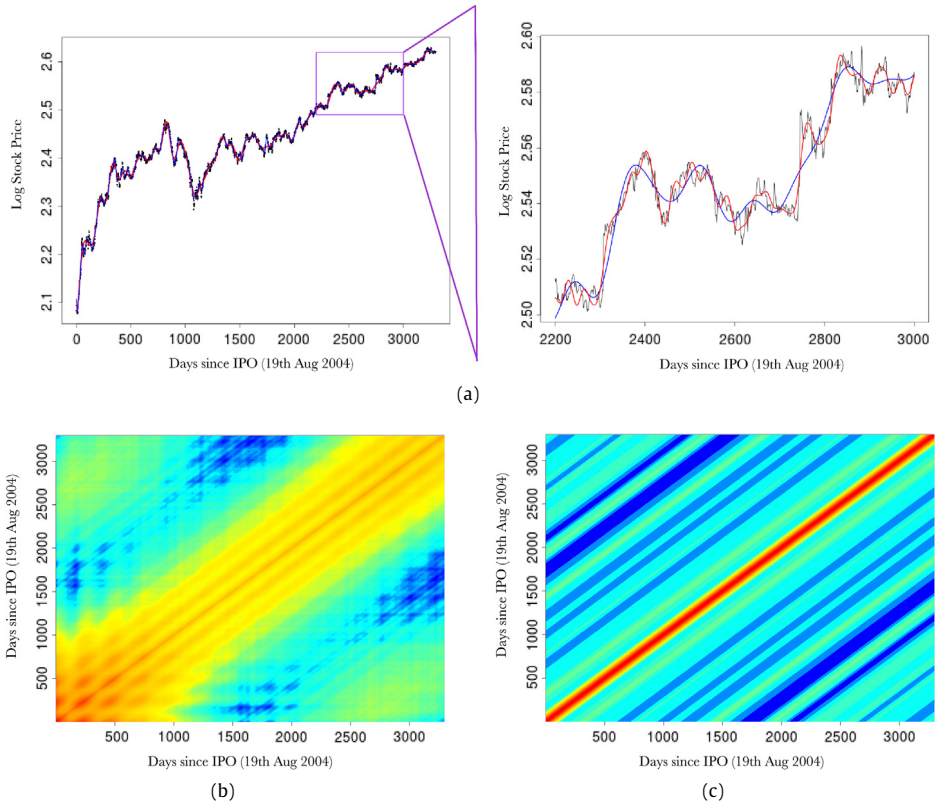


Fig. 3. (top) Log daily-high Google stock price (Y-axis), days since 19th August 2004 (X-axis). Vanilla random features (blue) our proposed approach (red), actual data (black), with a zoomed in transect (purple box); (bottom left) Image of covariance matrix for our nonstationary method; (bottom right) Image of covariance matrix for vanilla random Fourier features. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Fig. 6 shows our predicted surface (A) compared with the MRA (B), LatticeKrig (C) and SPDE (D) to the actual data (E). Our final mean absolute error point performance estimates were 3.5 for both our approach and MRA, 3.9 for the SPDE and 4.3 for LatticeKrig. CRPS scores and the posterior standard deviation for our method are shown in Fig. 5. Our model shows strong correspondence to the underlying data and highlights the suitability of using our approach in settings where no relevant covariates exist outside of the spatial coordinates. Our model is thus highly competitive with three of the best performing methods currently available (Heaton et al., 2017). Visual inspection indicates that while the RFF approach captures finer scale detail than both SPDE and LatticeKrig, it is unable to capture fine scale as well as the MRA approach, while having a comparable test score. The observation that RFFs fail to capture very fine scale details has, to our knowledge, never been reported before and deserves future research.

Fig. 7 shows 3 randomly sampled points and the covariance patterns around those points. For comparison Fig. 8 shows the equivalent plot when using an RFF stationary squared exponential kernel. In stark contrast to the stationary covariance function, which has an identical structure for all three points, the nonstationary kernel shows considerable heterogeneity in both patterns and shapes. Interestingly the learnt lengthscale/bandwidth seems to be much smaller in the stationary case than the nonstationary case, we hypothesise that this is due to the inability of the stationary kernel to learn the rich covariance structure needed to accurately model temperature anomaly. Intuitively,

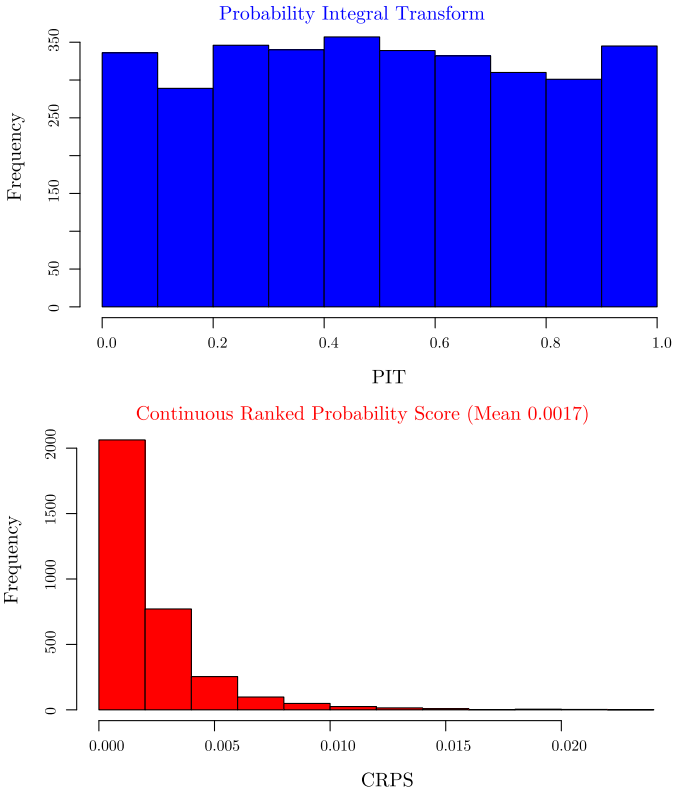


Fig. 4. Validation summary statistics for the google time series analysis with the probability integral transform (top) and the continuous ranked probability score (bottom). Both score statistics indicate a good fit for the posterior predictive distribution.

nonstationarity allows locally dependent covariance structures which conform to the properties of a particular location and imply (on average) larger similarity of nearby outputs and better generalisation ability. In contrast, stationary kernels are trying to fit one covariance structure to all locations and as a result end up with a much shorter lengthscale as it needs to apply to all directions from all locations. Our results are in concordance with other studies showing that temperature anomaly data is nonstationary (Remes et al., 2017; Samo and Roberts, 2015; Wu et al., 2007).

Of particular importance is that this interpolation problem can readily be expanded into multiple dimensions including time and other covariates. This is more challenging with the SPDE and MRA approach.

5. Discussion

We have shown that nonstationary kernels of arbitrary complexity are as easy to model as stationary ones, and can be learnt with sufficient efficiency to be applicable to datasets of all sizes. The qualitative superiority of predictions when using nonstationary kernels has previously been noted (Paciorek and Schervish, 2006). In many applications, such as in epidemiology where data can be noisy, generalisation accuracy is not the only measure of model performance, and there is a need for models that conform to known biological constraints and external field data. The flexibility of nonstationary kernels allows for more plausible realities to be modelled without the assumption of stationarity limiting the expressiveness of the predictions. It has also been noted that while nonstationary GPs give more sensible results than stationary GPs, they often show little

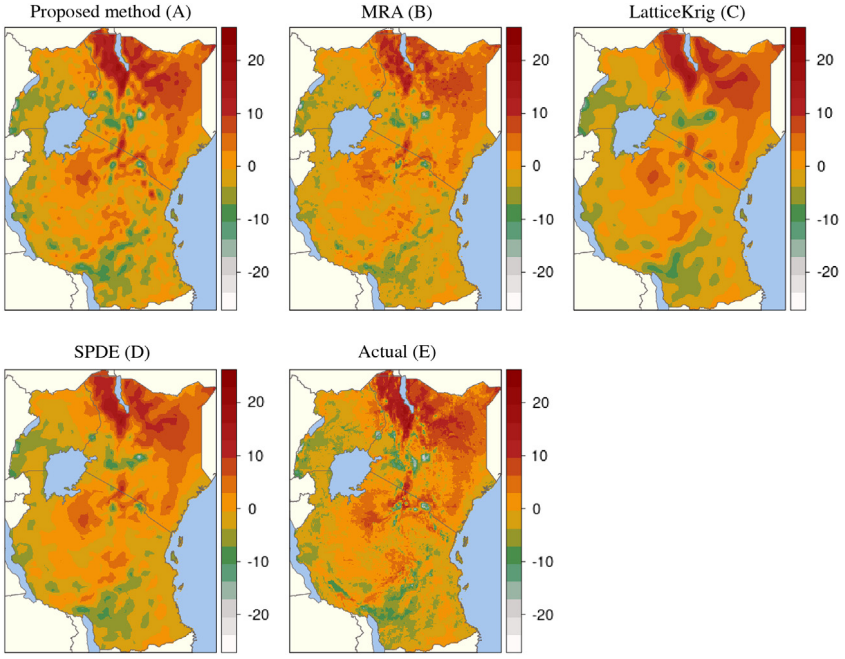


Fig. 5. Predicted surfaces of our approach and the current state-of-the-art: The SPDE, MRA and LatticeKrig.

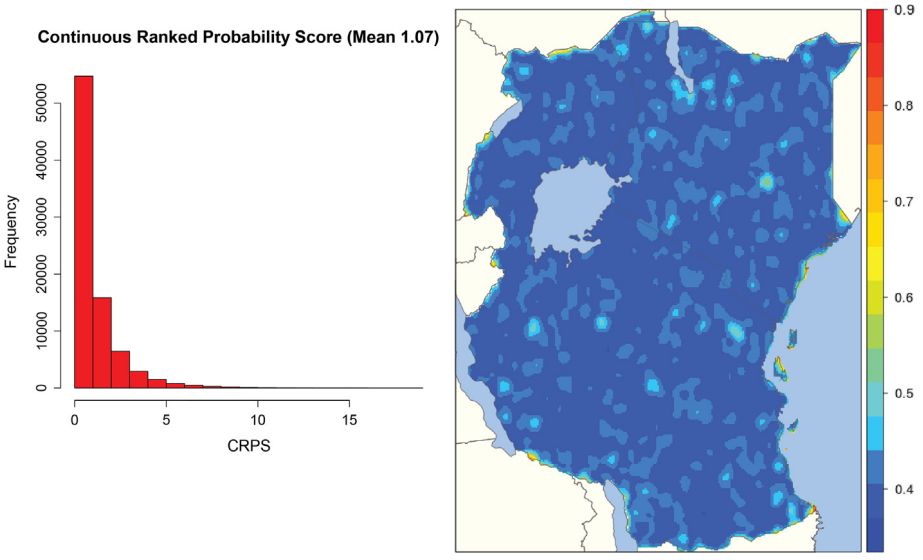


Fig. 6. CRPS scores and posterior standard deviation map using our proposed method for the temperature anomaly analysis.

generalisation improvement (Paciorek and Schervish, 2006). For the examples in this work we show clear improvements in generalisation accuracy when using nonstationary kernels. We conjecture that the differences in generalisation performance are likely due to the same reasons limiting neural

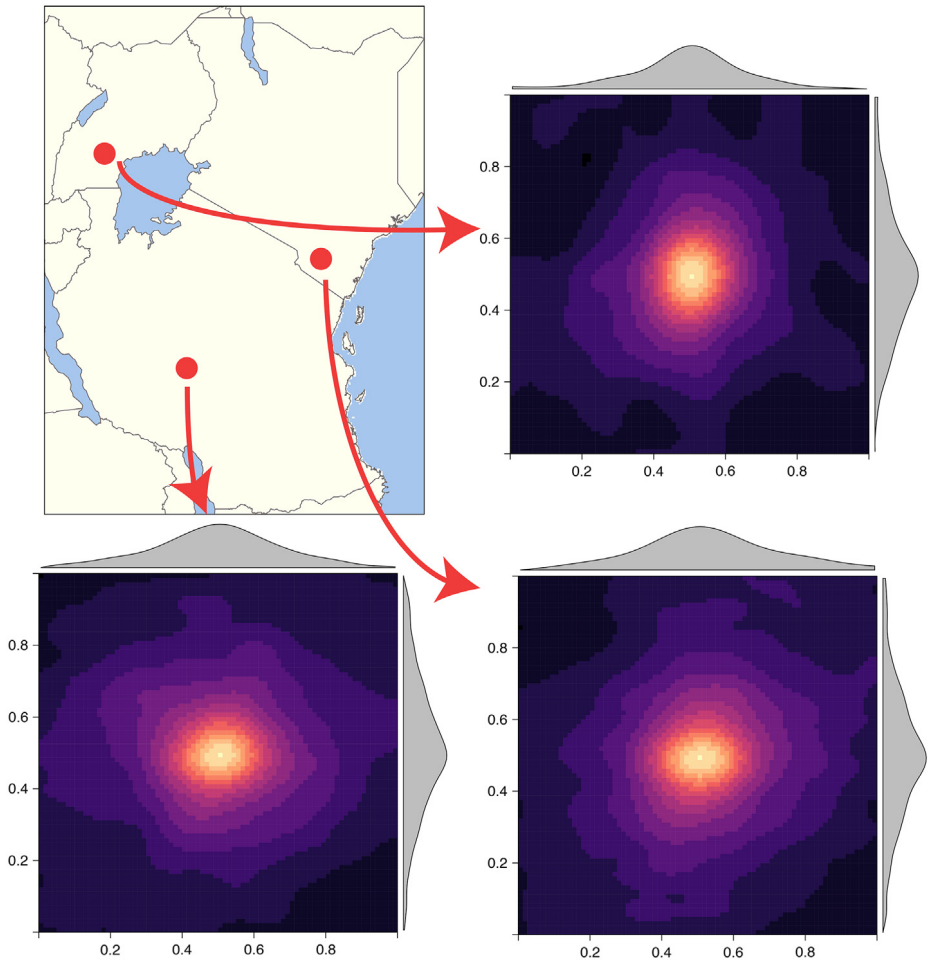


Fig. 7. Covariance matrix images for 3 random points showing different covariance structures due to nonstationarity.

network performance a decade ago (LeCun et al., 2015) - namely, a combination of small, poor quality data and a lack of generality in the underlying specification. Given more generalised specifications, such as those introduced in this paper, coupled with the current trend of increasing quantities of high quality data (Kambatla et al., 2014) we believe nonstationary approaches will be more and more relevant in spatio-temporal modelling.

There has long been codes of practice on which kernel to use on which spatial dataset (Diggle and Ribeiro, 2007) based on a priori assumptions about the roughness of the underlying process. Using the approach introduced in this paper, *ad hoc* choices of kernel and decisions on stationarity versus nonstationarity may no longer be needed as it may be possible to learn the kernel automatically from the data. For example, our approach can be easily modified to vary the degree of nonstationarity according to patterns found in the data.

In this work we have focused on optimising the marginal likelihood in Gaussian Process regression and added extra regularisation via Gaussian dropout. However, for non-Gaussian observation models the marginal likelihood cannot be obtained in a closed form. In these settings, one may resort to frequentist methods instead and resort to variational (Tan et al., 2013), approximate (Rue et al., 2009;

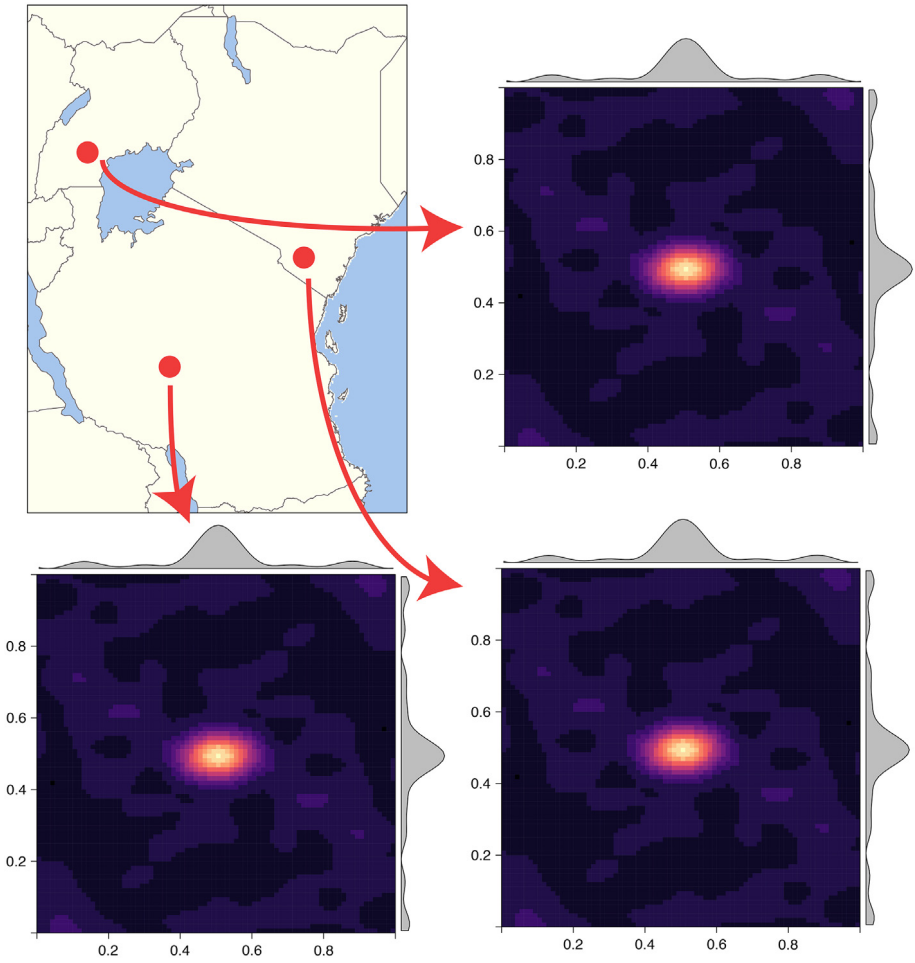


Fig. 8. Covariance matrix images for 3 random points showing identical covariance structures for all locations due to stationarity.

Minka, 2001) or suitable MCMC (Carpenter et al., 2017) approaches in order to provide uncertainty measures. For very large models with non-Gaussian observation models, stochastic gradient descent in mini-batches (Bengio, 2012) or stochastic gradient Bayesian methods (Chen et al., 2014) can be used. These Bayesian approaches also address the problem of hyperparameter choice for the Gaussian drop-out and the learning rate parameter.

As with all approximation methods there is a trade off (Bradley et al., 2016) using RFF's as opposed to other approaches. In general, data-dependent approaches have better convergence bounds better than data-independent approaches when there is a large gap in the eigen-spectrum of the kernel matrix (Yang et al., 2012). However, in contrast to low rank methods, RFFs approximate the entire kernel function directly via sampling from an explicit, lower dimensional, feature map which can improve performance over these methods in some settings (Stein, 2014). It should be noted that the theoretical properties and convergence bounds of RFFs are still not fully understood (Sriperumbudur and Szabo, 2015) and much more research is needed here; as it currently stands the uniform convergence bounds are similar to other approximation methods such as the Nyström (Wu et al., 2016), but simple modifications can greatly improve these bounds e.g. Yang et al. (2016), Yu et al.

(2016), Chang et al. (2017) and Chwialkowski et al. (2015). The benefits of RFFs lie in their scalability and ease of implementation coupled with their flexibility in kernel specification that is independent of dimension. In this work we demonstrate that an entire gambit of kernel complexity is possible within the same framework and that predictive performance can be improved in some settings with complex non-stationary kernels.

Author contributions

Conceived of and designed the research: SB, DS, SF. Drafted the manuscript: SB, DS, SF, JFT. Conducted the analyses: SB, JFT. Supported the analyses: SB, DS, SF, JFT. All authors discussed the results and contributed to the revision of the final manuscript.

Funding statement

SB is supported by the MRC outbreak centre and the Bill and Melinda Gates Foundation [OPP1152978]. DS would like to aknowledge the BigBayes group, MRC and EPSRC.

Appendix. Simple performance comparison to commonly used finite dimensional approaches

To provide intuition on the performance of RFFs versus other finite dimensional approximations we perform a simple simulation experiment. A full simulation analysis is beyond the scope of this work and we refer the reader to Bradley et al. (2016) and Heaton et al. (2017).

We simulate a full rank Gaussian process over a spatial two dimensional domain $X \in \mathbb{R}^2$ and evaluate the performance of 4 commonly used finite dimensional Gaussian process approximations including vanilla RFF's. The low rank approaches considered are (1) The Nyström Approximation (Williams and Seeger, 2001; Rasmussen and Williams, 2006), (2) Lindgren's stochastic partial differential equation approximation of Lindgren et al. (2011), (3) Higdon's Process convolutions (Higdon, 2002) and vanilla RFFs (Rahimi and Recht, 2008a). Approaches 1–3 were chosen because of their wide spread adoption in spatial applications (Diggle and Ribeiro, 2007; Bhatt et al., 2017, 2015; Giorgi and Diggle, 2017; Rasmussen and Williams, 2006), difference in approach, and ease of implementation.

1500 X locations were simulated uniformly over a 1×1 plane from a Matérn covariance function of smoothness $\nu = 1$ and variance $\sigma = 1$: $k_\theta(x_i, x_j) = \kappa/\tau \|x_i - x_j\| \mathcal{K}_1^{(2)}(\kappa \|x_i - x_j\|)$ with $\kappa = \sqrt{2}/\rho$ an inverse range parameter (for range, ρ), τ a precision (i.e., inverse variance) parameter, and $\mathcal{K}_1^{(2)}$ the modified Bessel function of the second kind and order 1. The range ρ, was then varied across $\rho \in \{\frac{1}{8}th, \frac{1}{15}th, \frac{1}{30}th, \frac{1}{50}th, \frac{1}{100}th, \frac{1}{200}th\}$ spanning a wide variety of plausible long and short range dependencies (Bhatt et al., 2015; Measure DHS, 2018).

For all approximations (1)–(4) the primal optimisation problem was solved (linear least squares regression) the mean squared error between the prediction y^* and the data y was calculated. Additionally for approximations (1)–(3) the Kullback–Leibler divergence between the full rank, covariance matrix Σ, and the finite dimensional covariance matrix Σ^* was calculated as $2KL(\Sigma, \Sigma^*) = tr((\Sigma^*)^{-1}\Sigma) - log(|\ \Sigma\ |) + log(|\ \Sigma^*\ |) - n$ (Stein, 2014). The mean squared error measured the approximation quality of the approximated Gaussian process evaluations, and the KL divergence measured the approximation quality of covariance matrix. Note the KL divergence was not calculated for (4) as the method outlined in Higdon (2002) is covariance matrix free.

Fig. 9 shows the results of this experiment. In terms of mean squared error the RFF, SPDE and Nyström approximations are all very similar. The process convolution method is not competitive with the other approaches unless ρ is small and the number of basis functions is large. In terms of KL divergence, the Nyström approximation performs best for all ranges followed by the RFF and SPDE approximations. This simple experiment demonstrates what has previously already been theoretically established; that the RFF approximation is competitive with other widely used covariance approximations (Sriperumbudur and Szabo, 2015; Li and Honorio, 2017; Rahimi and Recht, 2009, 2008a, b; Avron et al., 2017; Huang et al., 2014; Dai et al., 2014).

In the analysis performed, the computational complexity of (1–4) is not equal. The complexity of the RFF, SPDE and process convolutions is $\mathcal{O}(nm^3)$ for n data and m basis functions where as

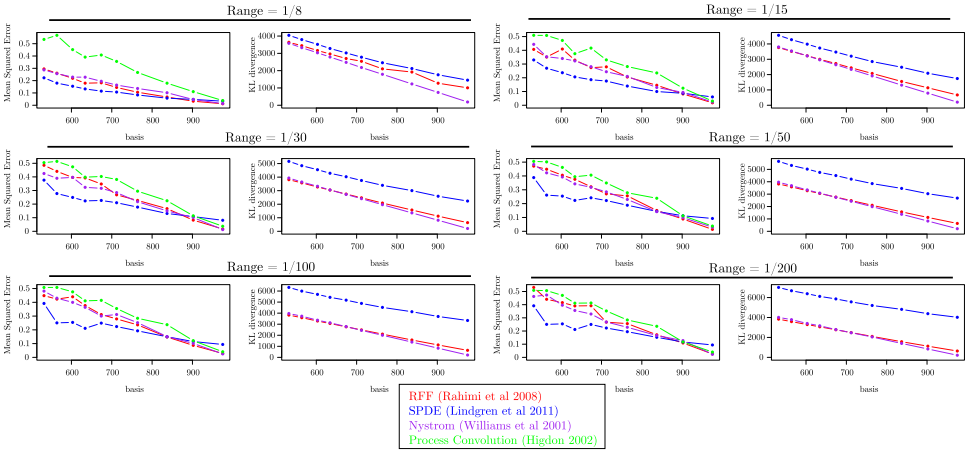


Fig. 9. Finite dimensional approximations applied to a full rank Gaussian process of various dependency ranges.

the complexity for the Nyström method is $\mathcal{O}(m^3)$ due to the need to perform an additional eigen decomposition step. Additionally, if sparsity was exploited then the SPDE's complexity can be as low as $\mathcal{O}(n + m^{\frac{3}{2}})$ which is far superior to the competing methods (Simpson et al., 2015). It should also be noted that the SPDE and process convolution methods are restricted low dimensional settings. Specifically the SPDE approach is restricted to $\mathbb{R}^2$ problems (and to Matérn, $\nu = 1$) and the curse of dimensionality practically limits the process convolution method. The RFF and Nyström methods can model high dimensional Gaussian processes without reductions in error due to dimension (Li and Honorio, 2017).

It should be noted that these experiments are performed using vanilla RFFs and simple adjustments such as performing integration by QMC (Yang et al., 2016), drawing features orthogonally by dimension (Yu et al., 2016), applying shrinkage via the Stein effect (Chang et al., 2017), or dampening the feature expansion with a function (Samo and Roberts, 2015; Remes et al., 2017; Chwialkowski et al., 2015) $h(x)$.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mane, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viegas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2016. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems, 3.

Avron, H., Kapralov, M., Musco, C.C.C., Musco, C.C.C., Velingker, A., Zandieh, A., 2017. Random fourier features for kernel ridge regression: Approximation bounds and statistical guarantees. In: Precup, D., Teh, Y.W. (Eds.), ICML. In: Proceedings of Machine Learning Research, vol. 70, PMLR, International Convention Centre, Sydney, Australia, pp. 253–262.

Baldi, P., Sadowski, P.J., 2013. Understanding Dropout.

Bengio, Y., 2012. Practical Recommendations for Gradient-Based Training of Deep Architectures. Springer, Berlin Heidelberg, pp. 437–478.

Bergstra, J., Bengio, Y., 2012. Random search for hyper-parameter optimization. J. Mach. Learn. Res. 13 (Feb), 281–305.

Berrocal, V.J., Raftery, A.E., Gneiting, T., Berrocal, V.J., Raftery, A.E., Gneiting, T., 2007. Combining spatial statistical and ensemble information in probabilistic weather forecasts. Mon. Weather Rev. 135 (4), 1386–1402.

Bhatt, S., Cameron, E., Flaxman, S.R., Weiss, D.J., Smith, D.L., Gething, P.W., 2017. Improved prediction accuracy for disease risk mapping using Gaussian Process stacked generalisation. J. R. Soc. Interface 14 (134), 20170520 https://www.ncbi.nlm.nih.gov/pubmed/28931634.

Bhatt, S., Gething, P.W., Brady, O.J., Messina, J.P., Farlow, A.W., Moyes, C.L., Drake, J.M., Brownstein, J.S., Hoen, A.G., Sankoh, O., Myers, M.F., George, D.B., Jaenisch, T., Wint, G.R.W., William Wint, G.R., Simmons, C.P., Scott, T.W., Farrar, J.J., Hay, S.I., 2013. The global distribution and burden of dengue. Nature 496 (7446), 504–507.

Bhatt, S., Weiss, D.J., Cameron, E., Bisanzio, D., Mappin, B., Dalrymple, U., Battle, K.E., Moyes, C.L., Henry, A., Eckhoff, P.A., Wenger, E.A., Briët, O., Penny, M.A., Smith, T.A., Bennett, A., Yukich, J., Eisele, T.P., Griffin, J.T., Fergus, C.A., Lynch, M., Lindgren, F., Cohen, J.M., Murray, C.L., Smith, D.L., Hay, S.I., Cibulskis, R.E., Gething, P.W., 2015. The effect of malaria control on Plasmodium falciparum in Africa between 2000 and 2015. Nature 526 (7572), 207–211.

Bradley, J.R., Cressie, N., Shi, T., 2016. A comparison of spatial predictors when datasets could be very large. Stat. Surv. 10, 100–131.

Carpenter, B., Gelman, A., Hoffman, M.D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., Riddell, A., 2017. Stan : A probabilistic programming language. J. Stat. Softw. 76 (1), 1–32.

Chang, W.C., Li, C.L., Yang, Y., Póczos, B., 2017. Data-driven random fourier features using Stein effect. In: IJCAI International Joint Conference on Artificial Intelligence, pp. 1497–1503.

Chen, T., Fox, E., Guestrin, C., 2014. Stochastic Gradient Hamiltonian Monte Carlo, 1.

Chwialkowski, K.P., Ramdas, A., Sejdinovic, D., Gretton, A., 2015. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R. (Eds.), Fast Two-Sample Testing with Analytic Representations of Probability Measures. In: Advances in Neural Information Processing Systems, vol. 28, Curran Associates, Inc., pp. 1981–1989.

Cressie, N., Johannesson, G., 2008. Fixed rank kriging for very large spatial data sets. J. R. Stat. Soc. Ser. B Stat. Methodol. 70 (1), 209–226.

Dai, B., Xie, B., He, N., Liang, Y., Raj, A., Balcan, M.-F.F., Song, L., 2014. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (Eds.), Scalable Kernel Methods Via Doubly Stochastic Gradients. In: Advances in Neural Information Processing Systems, vol. 27, Curran Associates, Inc., pp. 3041–3049.

Datta, A., Banerjee, S., Finley, A.O., Gelfand, A.E., 2016. Hierarchical nearest-neighbor gaussian process models for large geostatistical datasets. J. Amer. Statist. Assoc. 111 (514), 800–812.

Diggle, P., Ribeiro, P., 2007. Model-Based Geostatistics. Springer, New York.

Feuerverger, A., Mureika, R.A., 1977. The empirical characteristic function and its applications. Ann. Statist. 5 (1), 88–97.

Finkenstadt, B., Held, L., Isham, V., 2007. Statistical Methods for Spatio-Temporal Systems. Chapman & Hall/CRC, p. 286.

Furrer, R., Genton, M.G., Nychka, D., 2006. Covariance tapering for interpolation of large spatial datasets. J. Comput. Graph. Statist. 15 (3), 502–523.

Gal, Y., Turner, R., 2015. Improving the Gaussian process sparse spectrum approximation by representing uncertainty in frequency inputs. In: Proceedings of the 32nd International Conference on Machine Learning, ICML-15, p. 3.

Genest, C., MacKay, J., 1986. The joy of copulas: Bivariate distributions with uniform marginals. Amer. Statist. 40 (4), 280, 11.

Genton, M.G., Lázaro-Gredilla, M., Quiñonero-Candela, J., Rasmussen, C.E., Figueiras-Vidal, A.R., Lázaro-Gredilla, M., Quiñonero-Candela, J., Rasmussen, C.E., Figueiras-Vidal, A.R., 2001. Journal of machine learning research : JMLR. In: The Journal of Machine Learning Research. MIT Press.

Giorgi, E., Diggle, P.J., 2017. PrevMap : An R package for prevalence mapping. J. Stat. Softw. 78 (8).

Hastie, T., Tibshirani, R., Friedman, J.H., 2009. The Elements of Statistical Learning. Springer.

Hay, S.I., Battle, K.E., Pigott, D.M., Smith, D.L., Moyes, C.L., Bhatt, S., Brownstein, J.S., Collier, N., Myers, M.F., George, D.B., Gething, P.W., 2013. Global mapping of infectious disease. Philos. Trans. R. Soc. B.

Heaton, M.J., Datta, A., Finley, A., Furrer, R., Guhaniyogi, R., Gerber, F., Gramacy, R.B., Hammerling, D., Katzfuss, M., Lindgren, F., Nychka, D.W., Sun, F., Zammit-Mangion, A., 2017. Methods for analyzing large spatial data: A review and comparison, ArXiv e-Prints, 10.

Held, L., Schrödle, B., Rue, H., 2010. Posterior and cross-validatory predictive checks: A comparison of MCMC and INLA. In: Statistical Modelling and Regression Structures: Festschrift in Honour of Ludwig Fahrmeir. pp. 91–110.

Higdon, D., 2002. Space and space-time modeling using process convolutions. In: Quantitative Methods for Current Environmental Issues.

Huang, P.S., Avron, H., Sainath, T.N., Sindhwani, V., Ramabhadran, B., 2014. Kernel methods match deep neural networks on TIMIT. In: ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, pp. 205–209.

Kambatla, K., Kollias, G., Kumar, V., Grama, A., 2014. Trends in big data analytics. J. Parallel Distrib. Comput. 74 (7), 2561–2573.

Kang, E.L., Cressie, N., 2011. Bayesian inference for the spatial random effects model. J. Amer. Statist. Assoc. 106 (495), 972–983.

Katzfuss, M., 2017. A multi-resolution approximation for massive spatial datasets. J. Amer. Statist. Assoc. 112 (517), 201–214.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980, 12.

Lázaro-Gredilla, M., Quinonero-Candela, J., Rasmussen, C.E., Figueiras-Vidal, A.R., 2010. Sparse spectrum gaussian process regression. J. Mach. Learn. Res. 11, 1865–1881.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521 (7553), 436–444, 5.

Lee, J.D., Panageas, I., Piliouras, G., Simchowitz, M., Jordan, M.I., Recht, B., 2017. First-order methods almost always avoid saddle points, arXiv preprint arXiv:1710.07406, 10.

Li, Y.-J., Honorio, J., 2017. The error probability of random fourier features is dimensionality independent, Arxiv, vol. 710.09953v.

Lindgren, F., Rue, H., Lindström, J., 2011. An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach. J. R. Stat. Soc. Ser. B Stat. Methodol. 73 (4), 423–498.

Matheson, J.E., Winkler, R.L., 1976. Scoring rules for continuous probability distributions. Manage. Sci. 22 (10), 1087–1096.

Measure DHS, 2018. Demographic and health surveys modelled surfaces.

Micchelli, C.A., Xu, Y., Zhang, H., 2006. Universal Kernels. J. Mach. Learn. Res. 7, 2651–2667.

Minka, T.P., 2001. Expectation propagation for approximate Bayesian inference, pp. 362–369, 8.

Nychka, D., Bandyopadhyay, S., Hammerling, D., Lindgren, F., Sain, S., 2015. A multiresolution gaussian process model for the analysis of large spatial datasets. J. Comput. Graph. Statist..

Obled, C., Creutin, J.D., 1986. Some developments in the use of empirical orthogonal functions for mapping meteorological fields. J. Clim. Appl. Meteorol. 25 (9), 1189–1204.

Pace, R., Barry, R., Gilley, O.W., Sirmans, C.F., 2000. A method for spatial–temporal forecasting with an application to real estate prices. Int. J. Forecast. 16 (2), 229–246.
Paciorek, C.J., Schervish, M.J., 2006. Spatial modelling using a new class of nonstationary covariance functions. Environmetrics 17 (5), 483–506.
Prechelt, L., 1998. Early Stopping - But When? Springer, Berlin, Heidelberg, pp. 55–69.
Quiñonero-Candela, J., Rasmussen, C.E., 2005. A unifying view of sparse approximate Gaussian process regression. J. Mach. Learn. Res. 6 (Dec), 1939–1959.
Rahimi, A., Recht, B., 2008a. Random features for large-scale kernel machines. In: Advances in Neural Information Processing.
Rahimi, A., Recht, B., 2008b. Uniform approximation of functions with random bases. In: 46th Annual Allerton Conference on Communication, Control, and Computing.
Rahimi, A., Recht, B., 2009. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. Adv. Neural Inf. Process. Syst. 1 (1), 1–8.
Rasmussen, C.C.E., Williams, C.C.K.I., 2006. Gaussian Processes for Machine Learning, Vol. 14. MIT press, Cambridge, p. 248.
Remes, S., Heinonen, M., Kaski, S., 2017. Non-stationary spectral kernels, arXiv preprint arXiv:1705.08736, 5.
Rudi, A., Rosasco, L., 2016. Generalization Properties of Learning with Random Features.
Rue, H., Martino, S., Chopin, N., 2009. Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. J. R. Stat. Soc. Ser. B Stat. Methodol. 71 (2), 319–392, 4.
Rue, H., Tjelmeland, H., 2002. Fitting Gaussian Markov random fields to Gaussian fields. Scand. J. Stat. 29 (1), 31–49.
Samo, Y.-L.K., Roberts, S., 2015. Generalized Spectral Kernels, 6.
Simpson, D., Illian, J.B., Lindgren, F., Sørbye, S.H., Rue, H., 2015. Going off grid: Computationally efficient inference for log-Gaussian Cox processes. Biometrika 103 (1), 49–70.
Snelson, E., Ghahramani, Z., 2012. Variable noise and dimensionality reduction for sparse Gaussian processes, arXiv preprint arXiv:1206.6873, 6.
Sriperumbudur, B.K., Szabo, Z., 2015. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R. (Eds.), Optimal Rates for Random Fourier Features. In: Advances in Neural Information Processing Systems, vol. 28, Curran Associates, Inc., pp. 1144–1152.
Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: A simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15, 1929–1958.
Stein, M.L., 2014. Limitations on low rank approximations for covariance matrices of spatial data. Spat. Stat.
Tan, L.S.L., Ong, V.M.H., Nott, D.J., Jasra, A., 2013. Variational inference for sparse spectrum Gaussian process regression, arXiv preprint arXiv:1306.1999, 6.
Wan, Z., Zhang, Y., Zhang, Q., Li, Z.-l., 2002. Validation of the land-surface temperature products retrieved from Terra Moderate Resolution Imaging Spectroradiometer data. Remote Sens. Environ. 83, 163–180.
Weiss, D.J., Atkinson, P.M., Bhatt, S., Mappin, B., Hay, S.I., Gething, P.W., 2014a. An effective approach for gap-filling continental scale remotely sensed time-series. ISPRS J. Photogramm. Remote Sens..
Weiss, D.J., Bhatt, S., Mappin, B., Van Boeckel, T.P., Smith, D.L., Hay, S.I., Gething, P.W., 2014b. Air temperature suitability for Plasmodium falciparum malaria transmission in Africa 2000-2012: A high-resolution spatiotemporal prediction. Malar. J. 13 (1).
Weiss, D.J., Mappin, B., Dalrymple, U., Bhatt, S., Cameron, E., Hay, S.I., Gething, P.W., 2015. Re-examining environmental correlates of Plasmodium falciparum Malaria endemicity: A data-intensive variable selection approach. Malar. J.
Wikle, C.K., Cressie, N., 1999. A dimension-reduced approach to space-time Kalman filtering. Biometrika 86 (4), 815–829.
Wikle, C.K., Milliff, R.F., Nychka, D., Berliner, L.M., 2001. Spatiotemporal hierarchical bayesian modeling tropical ocean surface winds. J. Amer. Statist. Assoc. 96 (454), 382–397.
Williams, C., Seeger, M.W., 2001. Using the nystrom method to speed up kernel machines. In: NIPS Proceedings, Vol. 13, pp. 682–688.
Wilson, A.G., Adams, R.P., 2013. Gaussian process kernels for pattern discovery and extrapolation, pp. 1067–1075, arXiv preprint arXiv:1302.4245.
Wu, Z., Huang, N.E., Long, S.R., Peng, C.-K., 2007. On the trend, detrending, and variability of nonlinear and nonstationary time series. Proc. Natl. Acad. Sci. USA 104 (38), 14889–14894, 9.
Wu, L., Yen, I.E.H., William, C., Chen, J., 2016. Revisiting random binning feature: Fast convergence and strong parallelizability lingfeiwu. Acm Sigcdd 421–434.
Yaglom, A.M., 1987. Correlation Theory of Stationary and Related Random Functions. In: Springer Series in Statistics, Springer New York, New York, NY.
Yang, T., Li, Y.-F., Mahdavi, M., Jin, R., Zhou, Z.-H., 2012. Nystrom method vs random fourier features a theoretical and empirical comparison. Adv. Neural Inf. Process. Syst. 485–493.
Yang, J., Sindhwani, V., Avron, H., Mahoney, M., 2016. Quasi-Monte Carlo feature maps for shift-invariant kernels. J. Mach. Learn. Res. 17, 1–38.
Yang, Z., Wilson, A., Smola, A., Song, L., 2015. A la Carte – Learning Fast Kernels, 2.
Yu, F.X., Suresh, A.T., Choromanski, K., Holtmann-Rice, D., Kumar, S., 2016. Orthogonal Random Features, 10.