Bayesian Analysis (2017) **12**, Number 2, pp. 583–614

High-Dimensional Bayesian Geostatistics

Sudipto Banerjee*

Abstract. With the growing capabilities of Geographic Information Systems (GIS) and user-friendly software, statisticians today routinely encounter geographically referenced data containing observations from a large number of spatial locations and time points. Over the last decade, hierarchical spatiotemporal process models have become widely deployed statistical tools for researchers to better understand the complex nature of spatial and temporal variability. However, fitting hierarchical spatiotemporal models often involves expensive matrix computations with complexity increasing in cubic order for the number of spatial locations and temporal points. This renders such models unfeasible for large data sets. This article offers a focused review of two methods for constructing well-defined highly scalable spatiotemporal stochastic processes. Both these processes can be used as "priors" for spatiotemporal random fields. The first approach constructs a low-rank process operating on a lower-dimensional subspace. The second approach constructs a Nearest-Neighbor Gaussian Process (NNGP) that ensures sparse precision matrices for its finite realizations. Both processes can be exploited as a scalable prior embedded within a rich hierarchical modeling framework to deliver full Bayesian inference. These approaches can be described as model-based solutions for big spatiotemporal datasets. The models ensure that the algorithmic complexity has $\sim n$ floating point operations (flops), where n the number of spatial locations (per iteration). We compare these methods and provide some insight into their methodological underpinnings.

Keywords: Bayesian statistics, Gaussian process, low rank Gaussian process, Nearest Neighbor Gaussian process (NNGP), predictive process, sparse Gaussian process, spatiotemporal statistics.

1 Introduction

The increased availability of inexpensive, high speed computing has enabled the collection of massive amounts of spatial and spatiotemporal datasets across many fields. This has resulted in widespread deployment of sophisticated Geographic Information Systems (GIS) and related software, and the ability to investigate challenging inferential questions related to geographically-referenced data. See, for example, the books by Cressie (1993), Stein (1999), Moller and Waagepetersen (2003), Schabenberger and Gotway (2004), Gelfand et al. (2010), Cressie and Wikle (2011) and Banerjee et al. (2014) for a variety of statistical methods and applications.

This article will focus only on point-referenced data, which refers to data referenced by points with coordinates (latitude-longitude, Easting-Northing etc.). Modeling typically proceeds from a spatial or spatiotemporal process that introduces dependence

*UCLA Department of Biostatistics, 650 Charles E. Young Drive South, Los Angeles, CA 90095-1772, sudipto@ucla.edu

 DOI: 10.1214/17-BA1056R

among any finite collection of random variables from an underlying random field. For our purposes, we will consider the stochastic process as an uncountable set of random variables, say $\{w(\ell) : \ell \in \mathcal{L}\}$, over a domain of interest $\mathcal{L}$, which is endowed with a probability law specifying the joint distribution for any finite sample from that set. For example, in spatial modeling $\mathcal{L}$ is often assumed to be a subset of points in the Euclidean space $\Re^d$ (usually $d = 2$ or 3) or, perhaps, a set of geographic coordinates over a sphere or ellipsoid. In spatiotemporal settings $\mathcal{L} = \mathcal{S} \times \mathcal{T}$, where $\mathcal{S} \subset \Re^d$ is the spatial region, $\mathcal{T} \subset [0, \infty)$ is the time domain and $\ell = (s, t)$ is a space-time coordinate with spatial location $s \in \mathcal{S}$ and time point $t \in \mathcal{T}$ (see, e.g., Gneiting and Guttorp, 2010, for details).

Such processes are specified with a *covariance function* $K_\theta(\ell, \ell')$ that gives the covariance between $w(\ell)$ and $w(\ell')$ for any two points ℓ and ℓ' in $\mathcal{L}$. For any finite collection $\mathcal{U} = \{\ell_1, \ell_2, \ldots, \ell_n\}$ in $\mathcal{L}$, let $w_\mathcal{U} = (w(\ell_1), w(\ell_2), \ldots, w(\ell_n))^\top$ be the realizations of the process over $\mathcal{U}$. Also, for two finite sets $\mathcal{U}$ and $\mathcal{V}$ containing n and m points in $\mathcal{L}$, respectively, we define the $n \times m$ matrix $K_\theta(\mathcal{U}, \mathcal{V}) = \mathrm{Cov}(w_\mathcal{U}, w_\mathcal{V} \,|\, \theta)$, where the covariances are evaluated using $K_\theta(\cdot, \cdot)$. When $\mathcal{U}$ or $\mathcal{V}$ contains a single point, $K_\theta(\mathcal{U}, \mathcal{V})$ is a row or column vector, respectively. A valid spatiotemporal covariance function ensures that $K_\theta(\mathcal{U},\mathcal{U})$ is positive definite for any finite set $\mathcal{U}$. In geostatistics, we usually deal with a fixed set of points $\mathcal{U}$ and, if the context is clear, we write $K_\theta(\mathcal{U},\mathcal{U})$ simply as K_θ. A popular specification assumes $\{w(\ell) : \ell \in \mathcal{L}\}$ is a zero-centered Gaussian process written as $w(\ell) \sim GP(0, K_\theta(\cdot, \cdot))$, which implies that the $n \times 1$ vector $w = (w(\ell_1), w(\ell_2) \ldots, w(\ell_n))^\top$ is distributed as $N(0, K_\theta)$, where K_θ is the $n \times n$ covariance matrix with (i, j)-th element $K_\theta(\ell_i, \ell_j)$. Various characterizations and classes of valid spatial (and spatiotemporal) covariance functions can be found in Gneiting and Guttorp (2010), Cressie (1993), Stein (1999), Gelfand et al. (2010), Cressie and Wikle (2011) and Banerjee et al. (2014) and numerous references therein. The more common assumptions are of *stationarity* and *isotropy*. The former assumes that $K_\theta(\ell, \ell') = K_\theta(\ell - \ell')$ depends upon the coordinates only through their separation vector, while isotropy goes a step further and assumes the covariance is a function of the distance between them.

Spatial and spatiotemporal processes are conveniently embedded within Bayesian hierarchical models. The most common geostatistical setting assumes a response or dependent variable $y(\ell)$ observed at a generic point ℓ along with a $p \times 1$ ($p < n$) vector of spatially referenced predictors $x(\ell)$. Model-based geostatistical data analysis customarily envisions a spatial regression model,

$$y(\ell) = x^\top(\ell)\beta + w(\ell) + \epsilon(\ell) \,, \tag{1}$$

where β is the $p \times 1$ vector of slopes, and the residual from the regression is the sum of a spatial or spatiotemporal process, $w(\ell) \sim GP(0, K_\theta(\cdot, \cdot))$ capturing spatial and/or temporal association, and an independent process, $\epsilon(\ell)$ modeling measurement error or fine scale variation attributed to disturbances at distances smaller than the minimum observed separations in space and time. A Bayesian spatial model can now be constructed from (1) as

$$p(\theta, \beta, \tau) \times N(w \,|\, 0, K_\theta) \times N(y \,|\, X\beta + w, D_\tau) \,, \tag{2}$$

where $y = (y(\ell_1), y(\ell_2), \ldots, y(\ell_n))^\top$ is the $n \times 1$ vector of observed outcomes, X is the $n \times p$ matrix of regressors with i-th row $x^\top(\ell_i)$ and the noise covariance matrix

$D(\tau)$ represents measurement error or micro-scale variation and depends upon a set of variance parameters τ. A common specification is $D_\tau = \tau^2 I_n$, where τ^2 is called the "nugget." The hierarchy is completed by assigning prior distributions to β, θ and τ.

Bayesian inference can proceed by sampling from the joint posterior density in (2) using, for example, Markov chain Monte Carlo (MCMC) methods (see, e.g., Robert and Casella, 2004). A major computational bottleneck emerges from the size of K_θ in computing (2). Since θ is unknown, each iteration of the model fitting algorithm will involve decomposing or factorizing K_θ, which typically requires $\sim n^3$ floating point operations (flops). Memory requirements are of the order $\sim n^2$. These become prohibitive for large values of n when K_θ has no exploitable structure. Evidently, multivariate process settings, where $y(\ell)$ is a $q \times 1$ vector of outcomes, exacerbate the computational burden by a factor of q. For Gaussian likelihoods, one can integrate out the random effects w from (2). This reduces the parameter space to $\{\tau^2, \theta, \beta\}$, but one still needs to work with $K_\theta + \tau^2 I_n$, which is again $n \times n$. These settings are referred to as "big-n" or "high-dimensional" problems in geostatistics and are widely encountered in environmental sciences today.

As modern data technologies are acquiring and exploiting massive amounts of spatiotemporal data, modeling and inference for large spatiotemporal datasets are receiving increased attention. In fact, it is impossible to provide a comprehensive review of all existing methods for geostatistical models for massive spatial data sets; Sun et al. (2011) offer an excellent review for a number of methods for high-dimensional geostatistics. The ideas at the core of fitting models for large spatial and spatiotemporal data concern effectively solving positive definite linear systems such as $Ax = b$, where A is a covariance matrix. Thus one can use probability models to build computationally efficient covariance matrices. One approach is to approximate or model A with a covariance structure that can significantly reduce the computational burden. An alternative is to model A^{-1} itself with an exploitable structure so that the solution $A^{-1}b$ is available without computing the inverse. For full Bayesian inference, one also needs to ensure that the determinant of A is available easily.

We remark that when inferring about stochastic processes, it is also possible to work in the spectral domain. This rich, and theoretically attractive, option has been advocated by Stein (1999) and Fuentes (2007) and completely avoids expensive matrix computations. The underlying idea is to transform to the space of frequencies, construct a periodogram (an estimate of the spectral density), and exploit the Whittle likelihood (see, e.g., Whittle, 1954; Guyon, 1995) in the spectral domain as an approximation to the data likelihood in the original domain. The Whittle likelihood requires no matrix inversion so, as a result, computation is very rapid. In principle, inversion back to the original space is straightforward. However, there are practical impediments. First, there is discretization to implement a fast Fourier transform whose performance can be tricky over large irregular domains. Predictive inference at arbitrary locations also will not be straightforward. Other issues include arbitrariness to the development of a periodogram. Empirical experience is employed to suggest how many low frequencies should be discarded. Also, there is concern regarding the performance of the Whittle likelihood as an approximation to the exact likelihood. While this approximation is reasonably well centered, it does an unsatisfactory job in the tails (thus leading to

poor estimation of model variances). Lastly, modeling non-Gaussian first stages will entail unobservable random spatial effects, making the implementation impossible. In summary, use of the spectral domain with regard to handling large n, while theoretically attractive, has limited applicability.

Broadly speaking, model-based approaches for large spatial datasets proceeds from either exploiting "low-rank" models or exploiting "sparsity". The former attempts to construct Gaussian processes on a lower-dimensional subspace (see, e.g., Wikle and Cressie, 1999; Higdon, 2002a; Kammann and Wand, 2003; Quinoñero and Rasmussen, 2005; Stein, 2007; Gramacy and Lee, 2008; Stein, 2008; Cressie and Johannesson, 2008; Banerjee et al., 2008; Crainiceanu et al., 2008; Sansó et al., 2008; Finley et al., 2009a; Lemos and Sansó, 2009; Cressie et al., 2010) in spatial, spatiotemporal and more general Gaussian process regression settings. Sparse approaches include covariance tapering (see, e.g., Furrer et al., 2006; Kaufman et al., 2008; Du et al., 2009; Shaby and Ruppert, 2012) using compactly supported covariance functions. This is effective for parameter estimation and interpolation of the response ("kriging"), but it has not been fully evaluated for fully Bayesian inference on residual or latent processes. Introducing sparsity in K_θ^{-1} is prevalent in approximating Gaussian process likelihoods using Markov random fields (e.g., Rue and Held, 2005), products of lower dimensional conditional distributions (Vecchia, 1988, 1992; Stein et al., 2004), or composite likelihoods (e.g., Bevilacqua and Gaetan, 2014; Eidsvik et al., 2014).

This article aims to provide a focused review of some massively scalable Bayesian hierarchical models for spatiotemporal data. The aim is not to provide a comprehensive review of all existing methods. Instead, we focus upon two fully model-based approaches that can be easily embedded within hierarchical models and deliver full Bayesian inference. These are low-rank processes and sparsity-inducing processes. Both these processes can be used as "priors" for spatiotemporal random fields. Here is a brief outline of the paper. Section 2 discusses a Bayesian hierarchical framework for low-rank models and their implementation. Section 3 discusses some recent developments in sparsity-inducing Gaussian processes, especially nearest-neighbor Gaussian processes, and their implementation. Finally, Section 4 provides a brief account of outstanding issues for future research.

2 Hierarchical low-rank models

A popular way of dealing with large spatial datasets is to devise models that bring about dimension reduction (Wikle and Cressie, 1999). A *low rank* or *reduced rank* specification is typically based upon a representation or approximation in terms of the realizations of some latent process over a smaller set of points, often referred to as *knots*. To be precise,

$$w(\ell) \approx \tilde{w}(\ell) = \sum_{j=1}^{r} b_\theta(\ell, \ell_j^*) z(\ell_j^*) = b_\theta^\top(\ell) z, \tag{3}$$

where $z(\ell)$ is a well-defined process and $b_\theta(s, s')$ is a family of basis functions possibly depending upon some parameters θ. The collection of r locations $\{\ell_1^*, \ell_2^*, \ldots, \ell_r^*\}$ are the knots, $b_\theta(\ell)$ and z are $r \times 1$ vectors with components $b_\theta(\ell, \ell_j^*)$ and $z(\ell_j^*)$, respectively. For

any collection of n points, the $n \times 1$ vector $\tilde{w} = (\tilde{w}(\ell_1), \tilde{w}(\ell_2), \ldots, \tilde{w}(\ell_n))^\top$ is represented as $\tilde{w} = B_\theta z$, where B_θ is $n \times r$ with (i, j)-th element $b_\theta(\ell_i, \ell_j^*)$. Irrespective of how big n is, we now have to work with the r (instead of n) $z(\ell_j^*)$'s and the $n \times r$ matrix B_θ. Since we anticipate $r << n$, the consequential dimension reduction is evident and, since we will write the model in terms of the z's (with the $\tilde{w}$'s being deterministic from the z's, given $b_\theta(\cdot, \cdot)$), the associated matrices we work with will be $r \times r$. Evidently, $\tilde{w}(\ell)$ as defined in (3) spans only an r-dimensional space. When $n > r$, the joint distribution of $\tilde{w}$ is singular. However, we do create a valid stochastic process with covariance function

$$\text{cov}(\tilde{w}(\ell), \tilde{w}(\ell')) = b_\theta^\top(\ell) V_z b_\theta(\ell') \,, \tag{4}$$

where V_z is the variance-covariance matrix (also depends upon parameter θ) for z. From (4), we see that, even if $b_\theta(\cdot, \cdot)$ is stationary, the induced covariance function is not. If the z's are Gaussian, then $\tilde{w}(\ell)$ is a Gaussian process. Every choice of basis functions yields a process and there are too many choices to enumerate here. Wikle (2010) offers an excellent overview of low rank models.

Different families of spatial models emerge from different specifications for the process $z(\ell)$ and the basis functions $b_\theta(\ell, \ell')$. In fact, (3) can be used to construct classes of rich and flexible processes. Furthermore, such constructions need not be restricted to low rank models. If dimension reduction is not a concern, then full rank models can be constructed by taking $r = n$ basis functions in (3). A very popular specification for $z(\ell)$ is a white noise process so that $z \sim N(0, \sigma^2 I_n)$, whereupon (4) simplifies to $\sigma^2 b_\theta(\ell)^\top b_\theta(\ell')$. A natural choice for the basis functions is a kernel function, say $b_\theta(\ell, \ell') = K_\theta(\ell - \ell')$, which puts more weight on ℓ' near ℓ. Variants of this form have been called "moving average" models and explored by Barry and Ver Hoef (1996), while the term "kernel convolution" has been used in a series of papers by Higdon and collaborators (Higdon, 1998; Higdon et al., 1999; Higdon, 2002b) to not only achieve dimension reduction, but also model nonstationary and multivariate spatial processes. The kernel (which induces a parametric covariance function) can depend upon parameters and might even be spatially varying (Higdon, 2002b; Paciorek and Schervish, 2006). Sansó et al. (2008) use discrete kernel convolutions of independent processes to construct two different class of computationally efficient spatiotemporal processes.

Some choices of basis functions can be more computationally efficient than others depending upon the specific application. For example, Cressie and Johannesson (2008) (also see Shi and Cressie (2007)) discuss "Fixed Rank Kriging" (FRK) by constructing B_θ using very flexible families of non-stationary covariance functions to carry out high-dimensional kriging, Cressie et al. (2010) extend FRK to spatiotemporal settings calling the procedure "Fixed Rank Filtering" (FRF), Katzfuss and Cressie (2012) provide efficient constructions for B_θ for massive spatiotemporal datasets, and Katzfuss (2013) uses spatial basis functions to capture medium to long range dependence and tapers the residual $w(\ell) - \tilde{w}(\ell)$ to capture fine scale dependence. Multiresolution basis functions (see, e.g., Nychka et al., 2002, 2015) have been shown to be effective in building computationally efficient nonstationary models. These papers amply demonstrate the versatility of low-rank approaches using different basis functions.

A different approach is to specify the $z(\ell)$ as a spatial process model having a selected covariance function. This process is called the parent process and one can

derive a low-rank process $\tilde{w}(\ell)$ from the parent process. For example, one could use the Karhunen–Loeve (infinite) basis expansion for a Gaussian process (see, e.g., Rasmussen and Williams, 2005; Banerjee et al., 2014) and truncate it to a finite number of terms to obtain a low-rank process. Another example is to project the realizations of the parent process onto a lower-dimensional subspace, which yields the *predictive process* and its variants; see Section 2.2 for details.

The idea underlying low-rank dimension reduction is not dissimilar to Bayesian linear regression. For example, consider a simplified version of the hierarchical model in (2), where $\beta = 0$ and the process parameters $\{\theta, \tau\}$ are fixed. A low rank version of (2) is obtained by replacing w with $B_\theta z$, so the joint distribution is

$$N(z \,|\, 0, V_z) \times N(y \,|\, B_\theta z, D_\tau) \,, \tag{5}$$

where y is $n \times 1$, z is $r \times 1$, D_τ and V_z are positive definite matrices of sizes $n \times n$ and $r \times r$, respectively, and B_θ is $n \times r$. The low rank specification is accommodated using $B_\theta z$ and the prior on z, while D_τ (usually diagonal) has the residual variance components. By computing the marginal covariance matrix $\text{var}\{y\}$ in two ways (Lindley and Smith, 1972), one arrives at the well-known Sherman–Woodbury–Morrison formula

$$(D_\tau + B_\theta V_z B_\theta^\top)^{-1} = D_\tau^{-1} - D_\tau^{-1} B_\theta (V_z^{-1} + B_\theta^\top D_\tau^{-1} B_\theta)^{-1} B_\theta^\top D_\tau^{-1} \,. \tag{6}$$

The above formula reveals dimension reduction in terms of the marginal covariance matrix for y. If D_τ is easily invertible (e.g., diagonal), then the inverse of an $n \times n$ covariance matrix of the form $D_\tau + B_\theta V_z B_\theta^\top$ can be computed efficiently using the right-hand-side which only involves inverses of $r \times r$ matrices and D_τ^{-1}. A companion formula for (6) is that for the determinant,

$$\det(D_\tau + B_\theta V_z B_\theta^\top) = \det(V_z) \det(D_\tau) \det(V_z^{-1} + B_\theta^\top D_\tau^{-1} B_\theta) \,, \tag{7}$$

which shows that the determinant of the $n \times n$ matrix can be computed as a product of the determinants of two $r \times r$ matrices and that of D_τ.

In practical Bayesian computations, however, it is less efficient to directly use the formulas in (6) and (7). Since both the inverse and the determinant are needed, it is more useful to compute the Cholesky decomposition of the covariance matrix. In fact, one can avoid (6) completely and resort to a common trick in hierarchical models (see, e.g., Gelman et al., 2013) and smoothed analysis of variance (Hodges, 2013) that expresses (5) as the linear model

$$\underbrace{\begin{bmatrix} D_\tau^{-1/2} y \\ 0 \end{bmatrix}}_{y_*} = \underbrace{\begin{bmatrix} D_\tau^{-1/2} B_\theta \\ V_z^{-1/2} \end{bmatrix}}_{B_*} z + \underbrace{\begin{bmatrix} e_1 \\ e_2 \end{bmatrix}}_{e_*} \,, \quad \text{where } e_* \sim N(0, I_{n+r}) \,, \tag{8}$$

$V_z^{1/2}$ and $D_\tau^{1/2}$ are matrix square roots of V_z and D_τ, respectively. For example, in practice D_τ is diagonal so $D_\tau^{1/2}$ is simply the square root of the diagonal elements of D_τ, while $V_z^{1/2}$ is the triangular (upper or lower) Cholesky factor of the $r \times r$ matrix

V_z. The marginal density of $p(y_* \mid \theta, \tau)$ after integrating out z now corresponds to the linear model $y_* = B_*\hat{z} + e_*$, where $\hat{z}$ is the ordinary least-square estimate of z. Such computations are easily conducted in statistical programming environments such as `R` by applying the `chol` function to obtain the Cholesky factor $V_z^{1/2}$, a `backsolve` function to efficiently obtain $V_z^{-1/2}z$ in constructing (8), and an `lm` function to compute the least squares estimate of z using the QR decomposition of the design matrix B_*. We discuss implementation of low rank hierarchical models in a more general contexts in Section 2.3.

2.1 Biases in low-rank models

Irrespective of the precise specifications, low-rank models tend to underestimate uncertainty (since they are driven by a finite number of random variables), hence, overestimate the residual variance (i.e., the nugget). Put differently, this arises from systemic over-smoothing or model under-specification by the low-rank model when compared to the parent model. For example, if $w(\ell) = \tilde{w}(\ell) + \eta(\ell)$, where $w(\ell)$ is the parent process and $\tilde{w}(\ell)$ is a low-rank approximation, then ignoring the residual $\eta(\ell) = w(\ell) - \tilde{w}(\ell)$ can result in loss of uncertainty and oversmoothing. In settings where the spatial signal is weak compared to the noise, such biases will be less pronounced. Also, it is conceivable that in certain specific case studies proper choices of basis functions (e.g., multiresolution basis functions) will be able to capture much of the spatial behavior and the effect of the bias will be mitigated. However, in general it will be preferable to develop models that will be able to compensate for the overestimation of the nugget.

This phenomenon, in fact, is not dissimilar to what is seen in linear regression models and is especially transparent from writing the parent likelihood and low-rank likelihood as mixed linear models. To elucidate, suppose, without much loss of generality, that $\mathcal{U}$ is a set with n points of which the first r act as the knots. Let us write the Gaussian likelihood with the parent process as $N(y \mid Bu, \tau^2 I)$, where B is the $n \times n$ lower-triangular Cholesky factor of K_θ ($B = B_\theta$ depends on θ, but we suppress this here) and $u = (u_1, u_2, \ldots, u_n)^\top$ is now an $n \times 1$ vector such that $u_i \overset{iid}{\sim} N(0,1)$. Writing $B = [B_1 : B_2]$, where B_1 has $r < n$ columns, suppose we derive a low-rank model by truncating to only the first r basis functions. The corresponding likelihood is $N(y \mid B_1\tilde{u}_1, \tau^2 I)$, where $\tilde{u}_1$ is an $r \times 1$ vector whose components are independently and identically distributed $N(0,1)$ variables. Customary linear model calculations reveal that the magnitude of the residual vector from the parent model is given by $y^\top(I - P_B)y$, while that from the low-rank model is given by $y^\top(I - P_{B_1})y$, where P_A denotes the orthogonal projector matrix onto the column space of any matrix A. Using the fact that $P_B = P_{B_1} + P_{[(I-P_{B_1})B_2]}$, which is a standard result in linear model theory, we find the excess residual variability in the low-rank likelihood is summarized by $y^\top P_{[(I-P_{B_1})B_2]}y$ which can be substantial when r is much smaller than n.

In practical data analysis, the above phenomenon is usually manifested by an overestimation of the nugget variance as it absorbs the residual variation from the low-rank approximation. Consider the following simple experiment. We simulated a spatial dataset using the spatial regression model in (1) with $n = 200$ fixed spatial locations, say

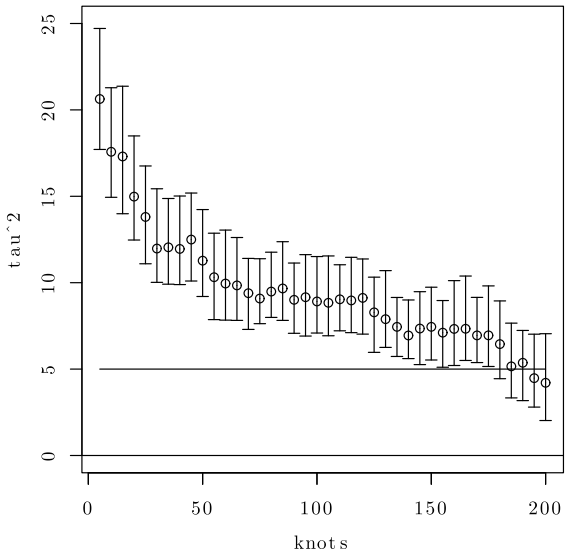


Figure 1: 95% credible intervals for the nugget for 40 different low-rank radial-basis models with knots varying between 5 and 200 in steps of 5. The horizontal line at $\tau^2 = 5$ denotes the true value of τ^2 with which the data was simulated.

$\{\ell_1, \ell_2, \ldots, \ell_n\}$, within the unit square, and setting $\beta = 0$, $\tau^2 = 5$, $w(\ell) \sim GP(0, K_\theta)$, where $K_\theta(\ell_i, \ell_j) = \sigma^2 \exp(-\phi\|\ell_i - \ell_j\|)$ with $\sigma^2 = 5$ and $\phi = 9$. We then fit the low rank model (5) with $D = \tau^2 I_{n\times n}$, $V = I_{r\times r}$, and B as the $n \times r$ matrix with i-th row $b^\top(\ell_i) = K_\theta(\ell_i, \mathcal{U}^*)K_\theta^{-1/2}(\mathcal{U}^*, \mathcal{U}^*)$, where $\mathcal{U}^* = \{\ell_1^*, \ell_2^*, \ldots, \ell_r^*\}$ is a set of r knots, $K_\theta(\ell_i, \mathcal{U}^*)$ is the $1 \times r$ vector with j-th element $K_\theta(\ell_i, \ell_j^*)$ and $K_\theta^{-1/2}(\mathcal{U}^*, \mathcal{U}^*)$ is the inverse of the lower-triangular Cholesky factor of the $r \times r$ matrix with elements $K_\theta(\ell_i^*, \ell_j^*)$. This emerges from using low-rank radial basis functions in (3); (see, e.g., Ruppert and Carroll, 2003). We fit 40 such models increasing r from 5 to 200 in steps of 5. Figure 1 presents the 95% posterior credible intervals for τ^2. Even with $r = 175$ knots for a dataset with just 200 spatial locations, the estimate of the nugget was significantly different from the true value of the parameter. This indicates that low rank processes may be unable to accurately estimate the nugget from the true process. Also, they will likely produce oversmoothed interpolated maps of the underlying spatial process and impair predictive performance. As one specific example, Table 4 in Banerjee et al. (2008) report less than optimal posterior predictive coverage from a predictive process model (see Section 2.2) with over 500 knots for a dataset comprising 15,000 locations.

Although this excess residual variability can be quantified as above (for any given value of the covariance parameters θ), it is less clear how the low-rank likelihood could be modified to compensate for this oversmoothing without adding significantly to the computational burden. Matters are complicated by the fact that expressions for the excess variability will involve the unknown process parameters θ, which must be estimated. In fact, not all low-rank models deliver a straightforward quantification for this bias.

For instance, low-rank models based upon kernel convolutions approximate $w(\ell)$ with $w_{KC}(\ell) = \sum_{j=1}^{n^*} K_\theta(\ell - \ell_j^*, \theta)u_j$, where $K_\theta(\cdot)$ is some kernel function and $u_j \overset{iid}{\sim} N(0,1)$, assumed to arise from a Brownian motion $U(\omega)$ on $\Re^2$. The difference $w(\ell) - w_{KC}(\ell)$ does not, in general, render a closed form and may be difficult to approximate efficiently.

2.2 Predictive process models and variants

One particular class of low-rank processes have been especially useful in providing easy tractability to the residual process. Let $w(\ell) \sim GP(0, K_\theta(\cdot,\cdot))$ and let w^* be the $r \times 1$ vector of $w(\ell_j^*)$'s over a set $\mathcal{U}^*$ of r knots. The usual spatial interpolant (that leads to "kriging") at an arbitrary site ℓ is

$$\tilde{w}(\ell) = \mathrm{E}[w(\ell) \,|\, w^*] = K_\theta(\ell, \mathcal{U}^*)K_\theta^{-1}(\mathcal{U}^*, \mathcal{U}^*)w^* \,. \tag{9}$$

This single site interpolator, in fact, is a well-defined process $\tilde{w}(\ell) \sim GP(0, \tilde{K}_\theta(\cdot,\cdot))$ with covariance function, $\tilde{K}_\theta(\ell,\ell') = K_\theta(\ell;\mathcal{U}^*)K_\theta^{-1}(\mathcal{U}^*,\mathcal{U}^*)K_\theta(\mathcal{U}^*,\ell')$. We refer to $\tilde{w}(\ell)$ as the *predictive process* derived from the *parent process* $w(\ell)$. The realizations of $\tilde{w}(\ell)$ are precisely the kriged predictions conditional upon a realization of $w(\ell)$ over $\mathcal{U}^*$. The process is completely specified given the covariance function of the parent process and the set of knots, $\mathcal{U}^*$. The corresponding basis functions in (3) are given by $b_\theta^\top(\ell) = K_\theta(\ell,\mathcal{U}^*)K_\theta^{-1}(\mathcal{U}^*,\mathcal{U}^*)$. These methods have are referred to as subset of regressors in Gaussian process regressions for large data sets in machine learning (Quinoñero and Rasmussen, 2005; Rasmussen and Williams, 2005). Banerjee et al. (2008) coined the term predictive process (as the process could be derived from kriging equations) and developed classes of scalable Bayesian hierarchical spatial process models by replacing the parent process with its predictive process counterpart. An alternate derivation is available by truncating the Karhunen–Loeve (infinite) basis expansion for a Gaussian process to a finite number of terms and solving (approximately) the integral eigensystem equation for $K_\theta(\ell,\ell')$ by an approximate linear system over the set of knots (see, e.g., Rasmussen and Williams, 2005; Sang and Huang, 2012; Banerjee et al., 2014).

Exploiting elementary properties of conditional expectations, we obtain

$$\mathrm{var}\{w(\ell)\} = \mathrm{var}\{\mathrm{E}[w(\ell) \,|\, w^*]\} + \mathrm{E}\{\mathrm{var}[w(\ell) \,|\, w^*]\} \geq \mathrm{var}\{\mathrm{E}[w(\ell) \,|\, w^*]\} \,, \tag{10}$$

which implies that $\mathrm{var}\{w(\ell)\} \geq \mathrm{var}\{\tilde{w}(\ell)\}$ and the variance of $\eta(\ell) = w(\ell) - \tilde{w}(\ell)$ is simply the difference of the variances. For Gaussian processes, we get the following closed form for $\mathrm{Cov}\{\eta(\ell), \eta(\ell')\}$,

$$K_{\eta,\theta}(\ell,\ell') = K_\theta(\ell,\ell') - K_\theta(\ell,\mathcal{U}^*)K_\theta^{-1}(\mathcal{U}^*,\mathcal{U}^*)K_\theta(\mathcal{U}^*,\ell') \,. \tag{11}$$

Therefore, $\mathrm{var}\{\eta(\ell)\} = K_{\eta,\theta}(\ell,\ell)$, which we denote as $\delta^2(\ell)$.

Perhaps the simplest way to remedy the bias in the predictive process is to approximate the residual process $\eta(\ell)$ with a heteroskedastic process $\tilde{\epsilon}(\ell) \overset{ind}{\sim} N(0, \delta^2(\ell))$. We construct a *modified* or *bias-adjusted* predictive process as

$$\tilde{w}_\epsilon(\ell) = \tilde{w}(\ell) + \tilde{\epsilon}(\ell) \,, \tag{12}$$

	μ	σ^2	τ^2	RMSPE
True	1	1	1	
$m = 49$				
PP	1.37 (0.29,2.61)	1.37 (0.65,2.37)	1.18 (1.07,1.23)	1.21
MPP	1.36 (0.51,2.39)	1.04 (0.52,1.92)	0.94 (0.68.1,14)	1.20
$m = 144$				
PP	1.36 (0.52,2.32)	1.39 (0.76,2.44)	1.09 (0.96, 1.24)	1.17
MPP	1.33 (0.50,2.24)	1.14 (0.64,1.78)	0.93 (0.76,1.22)	1.17
$m = 900$				
PP	1.31 (0.23, 2.55)	1.12 (0.85,1.58)	0.99 (0.85,1.16)	1.17
MPP	1.31 (0.23,2.63)	1.04 (0.76,1.49)	0.98 (0.87,1.21)	1.17

Table 1: Parameter estimates for the predictive process (PP) and modified predictive process (MPP) models in the univariate simulation.

where $\tilde{\epsilon}(\ell)$ is independent of $\tilde{w}(\ell)$. It is easy to see that $\text{var}\{\tilde{w}_\epsilon(\ell)\} = \text{var}\{w(\ell)\}$, so the variance of the two processes are the same. Also, the remedy is computationally efficient – adding an independent space-varying nugget does not incur substantial computational expense. Finley et al. (2009b) offer computational details for the modified predictive process, while Banerjee et al. (2010) show the effectiveness of the bias adjustment in mitigating the effect exhibited in Figure 1 and in estimating multiple variance components in the presence of different structured random effects.

We present a brief simulation example revealing the benefits of the modified predictive process. We generate 2000 locations within a $[0, 100] \times [0, 100]$ square and then generate the outcomes from (1) using only an intercept as the regressor, an exponential covariance function with range parameter $\phi = 0.06$ (i.e., such that the spatial correlation is ~ 0.05 at 50 distance units), scale $\sigma^2 = 1$ for the spatial process, and with nugget variance $\tau^2 = 1$. We then fit the predictive process and modified predictive process models derived from (1) using a hold out set of randomly selected sites, along with a separate set of regular lattices for the knots ($m = 49$, 144 and 900). Table 1 shows the posterior estimates and the square roots of mean squared predictive error (RMSPE) based on the predictions for the hold-out data. We clearly see the overestimation of τ^2 by the predictive process and that the modified predictive process is able to adjust for the τ^2. Not surprisingly, the RMSPE is essentially the same under either process model.

Further enhancements to the modified predictive process are possible. Since the modified predictive process adjusts only the variance, information in the covariance induced by the residual process $\eta(\ell)$ is lost. One alternative is to use the so called "full scale approximation" proposed by Sang et al. (2011) and Sang and Huang (2012), where $\eta(\ell)$ is approximated by a tapered process, say $\eta_{tap}(\ell)$. The covariance function for $\eta(\ell)$ is of the form $K_{\eta,\theta}(\ell, \ell')K_{tap,\nu}(\|\ell - \ell'\|)$, where $K_{\eta,\theta}(\ell, \ell')$ is as in (11) and $K_{tap,\nu}(\|\ell - \ell'\|)$ is a compactly supported covariance function that equals 0 beyond a distance ν (see, e.g., Furrer et al., 2006, for some practical choices.). This full scale approximation is also able to more effectively capture small scale dependence. Katzfuss (2013) extended some of these ideas by modeling the spatial error as a combination

of a low-rank component designed to capture medium to long-range dependence and a tapered component to capture local dependence.

Perhaps the most promising use of the predictive process, at least in terms of scalability to massive spatial datasets, is the recent multiresolution approximation proposed by Katzfuss (2017). Instead of approximating the residual process $\eta(\ell)$ in one step, the idea here is to partition the spatial domain recursively and construct a sequence of approximations. We start by partitioning the domain of interest $\mathcal{L}$ into J non-intersecting subregions, say $\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_J$, such that $\mathcal{L} = \cup_{j=1}^{J} \mathcal{L}_j$. We call the $\mathcal{L}_j$'s level-1 subregions. We fix a set of knots in $\mathcal{L}$ and write the parent process as $w(\ell) = \tilde{w}(\ell) + \eta(\ell)$, where $\tilde{w}(\ell)$ is the predictive process as in (9) and $\eta(\ell)$ is the residual Gaussian process with covariance function given by (11). At resolution 1, we replace $\eta(\ell)$ with a block-independent process $\eta_1(\ell)$ such that $\mathrm{Cov}\{\eta_1(\ell), \eta_1(\ell')\} = 0$ if ℓ and ℓ' are not in the same subregion and is equal to (11) if ℓ and ℓ' are in the same subregion.

At the second resolution, each $\mathcal{L}_j$ is partitioned into a set of disjoint subregions $\mathcal{L}_{j1}, \mathcal{L}_{j2}, \ldots, \mathcal{L}_{jm}$. We call these the level-2 subregions and choose a set of knots within each. We approximate $\eta_1(\ell) \approx \tilde{\eta_1}(\ell) + \eta_2(\ell)$, where $\tilde{\eta_1}(\ell)$ is the predictive process derived from $\eta_1(\ell)$ using the knots in $\mathcal{L}_j$ if $\ell \in \mathcal{L}_j$ and $\eta_2(\ell)$ is the analogous block-independent approximation across the subregions within each $\mathcal{L}_j$. Thus, $\mathrm{Cov}\{\eta_2(\ell), \eta_2(\ell')\} = 0$ if ℓ and ℓ' are not in the same level-2 subregion and will equal $\mathrm{Cov}\{\eta_1(\ell), \eta_1(\ell')\}$ when ℓ and ℓ' are in the same level-2 subregion. At resolution 3 we partition each of the level-2 subregions into level-3 subregions and continue the approximation of the residual process from the predictive process. At the end of M resolutions, we arrive at the mult-resolution predictive process $\tilde{w}_M(\ell) = \tilde{w}(\ell) + \sum_{i=1}^{M-1} \tilde{\eta}_i(\ell) + \eta_M(\ell)$, which, by construction, is a valid Gaussian process. The computational complexity with the multi-resolution predictive process is $\sim O(nM^2r^2)$, where M is the number of resolutions and r is the number of knots chosen within each subregion.

To summarize, we do not recommend the use of *just* a reduced/low rank model. To improve performance, it is necessary to approximate the residual process and, in this regard, the predictive process is especially attractive since the residual process is available explicitly.

2.3 Bayesian implementation for low-rank models

A very rich and flexible class of spatial and spatiotemporal models emerge from the hierarchical linear mixed model

$$p(\theta) \times p(\tau) \times N(\beta \,|\, \mu_\beta, V_\beta) \times N(z \,|\, 0, V_{z,\theta}) \times N(y \,|\, X\beta + B_\theta z, D_\tau) \;, \tag{13}$$

where y is an $n \times 1$ vector of possibly irregularly located observations, X is a known $n \times p$ matrix of regressors ($p < n$), $V_{u,\theta}$ and D_τ are families of $r \times r$ and $n \times n$ covariance matrices depending on unknown process parameters θ and τ, respectively, and B_θ is $n \times r$ with $r \leq n$. The low-rank models in (3) emerge when $r << n$ and B_θ is the matrix obtained by evaluating the basis functions. Proper prior distributions $p(\theta)$ and $p(\tau)$ for θ and τ, respectively, complete the hierarchical specification.

Bayesian inference proceeds, customarily, by sampling $\{\beta, z, \theta, \tau\}$ from (13) using Markov chain Monte Carlo (MCMC) methods. For faster convergence, we integrate out z from the model and first sample from $p(\theta, \tau, \beta \mid y) \propto p(\theta) \times p(\tau) \times N(\beta \mid \mu_\beta, V_\beta) \times N(y \mid X\beta, \Sigma_{y \mid \theta,\tau})$, where $\Sigma_{y \mid \theta,\tau} = B_\theta V_{z,\theta} B_\theta^\top + D_\tau$. Working directly with $\Sigma_{y \mid \theta,\tau}$ will be expensive. Usually D_τ is diagonal or sparse, so the expense is incurred from the matrix $B_\theta V_{z,\theta} B_\theta^\top$. Assuming that B_θ and $V_{z,u}$ are computationally inexpensive to construct for each θ and τ, $B_\theta V_{z,\theta} B_\theta^\top$ requires $\sim O(rn^2)$ flops. Using the Sherman–Woodbury–Morrison formula in (6) will avoid constructing $B_\theta V_{z,\theta} B_\theta^\top$ or inverting any $n \times n$ matrix. However, in practice it is better to not directly compute the right hand side of (6) as it involves some redundant matrix multiplications. Furthermore, we wish to obtain the determinant of $\Sigma_{y \mid \theta,\tau}$ cheaply. These are efficiently accomplished as outlined below.

The primary computational bottleneck lies in evaluating the multivariate Gaussian likelihood $N(y \mid X\beta, \Sigma_{y \mid \theta,\tau})$ which is required for updating the parameters $\{\theta, \tau\}$ (e.g., using random-walk Metropolis or Hamiltonian Monte Carlo steps). We can accomplish this effectively using two functions: $L = \texttt{chol}(V)$ which computes the Cholesky factorization for any positive definite matrix $V = LL^\top$, where L is lower-triangular, and $W = \texttt{trsolve}(T, B)$ which solves the triangular system $TW = B$ for a triangular (lower or upper) matrix T. We first compute

$$\left(B_\theta V_{z,\theta} B_\theta^\top + D_\tau\right)^{-1} = D_\tau^{-1/2}(I - H^\top H)D_\tau^{-1/2} \,, \tag{14}$$

where H is obtained by first computing $W = D^{-1/2}B_\theta$, then the Cholesky factorization $L = \texttt{chol}(V_{z,\theta}^{-1} + W^\top W)$, and finally solve the triangular system $H = \texttt{trsolve}(L, W^\top)$. Having obtained H, we compute $e = y - X\beta$, $m_1 = D^{-1/2}e$, $m_2 = Hm_1$, and obtain $T = \texttt{chol}(I_r - HH^\top)$. The log-target density for $\{\theta, \tau\}$ is then computed as

$$\log p(\theta) + \log p(\tau) - \frac{1}{2}\sum_{i=1}^{n} d_{ii} + \sum_{i=1}^{r} \log t_{ii} - \frac{1}{2}(m_1^\top m - m_2^\top m_2) \,, \tag{15}$$

where d_{ii}'s and t_{ii}'s are the diagonal elements of D_τ and T, respectively. The total number of flops required for evaluating the target is $O(nr^2 + r^3) \approx O(nr^2)$ (since $r << n$) which is considerably cheaper than the $O(n^3)$ flops that would have been required for the analogous computations in a full Gaussian process model. In practice, Gaussian proposal distributions are employed for the Metropolis algorithm and all parameters with positive support are transformed to their logarithmic scale. Therefore, the necessary Jacobian adjustments are made to (15) by adding some scalar quantities with negligible computational costs.

Starting with initial values for all parameters, each iteration of the MCMC executes the above calculations to provide a sample for $\{\theta, \tau\}$. The regression parameter β is then sampled from its full conditional distribution. Writing $\Sigma_y = B_\theta V_{z,\theta} B_\theta^\top + D_\tau$ as in (14), the full conditional distribution for β is $N(Aa, A)$, where $A^{-1} = \Sigma_\beta^{-1} + X^\top \Sigma_y^{-1} X$ and $a = \Sigma_\beta^{-1}\mu_\beta + X^\top \Sigma_y^{-1} y$. These are efficiently computed as $[f : F] = D^{-1/2}[y : X]$, $\tilde{F} = HF$ and setting $a = \Sigma_\beta^{-1}\mu_\beta + F^\top f - \tilde{F}^\top Hf$ and $L = \texttt{chol}(\Sigma_\beta^{-1} + F^\top F - \tilde{F}^\top \tilde{F})$. We then compute $\beta = \texttt{trsolve}(L^\top, \texttt{trsolve}(L, a)) + \texttt{trsolve}(L, \tilde{Z})$, where $\tilde{Z}$ is a conformable vector of independent $N(0, 1)$ variables.

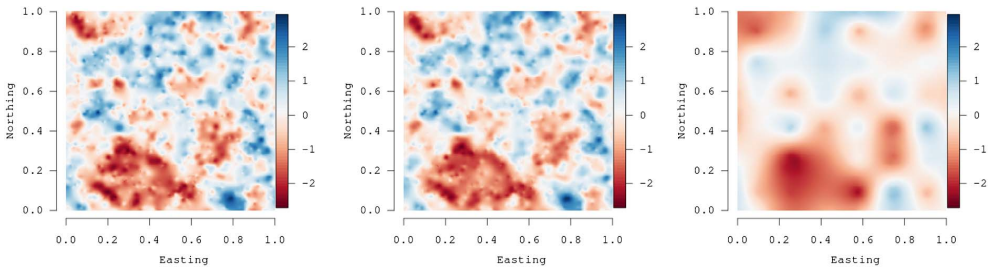
We repeat the above computations for each iteration of the MCMC algorithm using the current values of the process parameters in Σ_y. The algorithm described above will produce, after convergence, posterior samples for $\Omega = \{\theta, \tau, \beta\}$. We then sample from the posterior distribution $p(z \mid y) = \int p(z \mid \Omega, y)p(\Omega \mid y)d\Omega$, where $p(z \mid \Omega, y) = N(z \mid Aa, A)$ with $A = (V_{z,\theta}^{-1} + B_\theta^\top D_\tau^{-1} B_\theta)^{-1}$ and $a = B_\theta^\top D_\tau^{-1}(y - X\beta)$. For each Ω drawn from $p(\Omega \mid y)$ we will need to draw a corresponding z from $N(z \mid Aa, A)$. This will involve $\texttt{chol}(A)$. Since the number of knots r is usually fixed at a value much smaller than n, obtaining $\texttt{chol}(A)$ is $\sim O(r^3)$ and not as expensive. However, it will involve the inverse of $V_{z,\theta}$, which is computed using $\texttt{chol}(V_{z,\theta})$ and can be numerically unstable for certain smoother covariance functions such as the Gaussian or the Matérn with large ν. A numerically more stable algorithm exploits the relation $A = Q - Q(V_{z,\theta} + Q)^{-1}Q$, where $Q^{-1} = B_\theta^\top D_\tau^{-1} B_\theta$. For each Ω sampled from $p(\Omega \mid y)$, we compute $L = \texttt{chol}(V_{z,\theta} + Q)$, $W = \texttt{trsolve}(L, Q)$ and $L = Q - W^\top W$. We generate an $r \times 1$ vector $Z^* \sim N(0, I_r)$ and set $z = L(Z^* + L^\top a)$. Repeating this for each Ω drawn from $p(\Omega \mid y)$ produces a sample of z's from $p(z \mid y)$.

Finally, we seek predictive inference for $y(\ell_0)$ at any arbitrary space-time coordinate ℓ_0. Given $x^\top(\ell_0)$, we draw $y(\ell_0) \sim N\left(x^\top(\ell_0)\beta + b_\theta^\top(\ell_0)z, \tau^2\right)$ for every posterior sample of Ω and z. This yields the corresponding posterior predictive samples for $z(\ell_0)$ and $y(\ell_0)$. Posterior predictive samples of the latent processes can also be easily computed as $z(\ell_0) = b_\theta^\top(\ell_0)z$ for each posterior sample of the z and θ. Posterior predictive distributions at any of the observed ℓ_i's yield *replicated* data (see, e.g., Gelman et al., 2013) that can be used for model assessment and comparisons. Finley et al. (2015) provide more extensive implementation details for models such as (13) in the context of the `spBayes` package in `R`.

3 Sparsity-inducing nearest-neighbor Gaussian processes

Low-rank models have been, and continue to be, widely employed for analyzing spatial and spatiotemporal data. The algorithmic cost for fitting low-rank models typically decrease from $O(n^3)$ to $O(nr^2 + r^3) \approx O(nr^2)$ flops since $n >> r$. However, when n is large, empirical investigations suggest that r must be fairly large to adequately approximate the parent process and the nr^2 flops become exorbitant. Furthermore, low-rank models can perform poorly depending upon the smoothness of the underlying process or when neighboring observations are strongly correlated and the spatial signal dominates the noise (Stein, 2014).

As an example, consider part of the simulation experiment presented in Datta et al. (2016a), where a spatial random field was generated over a unit square using a Gaussian process with fixed spatial process parameters over a set of 2500 locations. We then fit a full Gaussian process model and a predictive process model with 64 knots. Figure 2 presents the results (see, e.g., Datta et al., 2016a, for details.) While the estimated random field from the full Gaussian process is almost indistinguishable from the true random field, the surface obtained from the predictive process with 64 locations substantially oversmooths. This oversmoothing can be ameliorated by using a larger number of knots, but that adds to the computational burden.



(a) True w (b) Full GP (c) PPGP 64 knots

Figure 2: Comparing estimates of a simulated random field using a full Gaussian Process (Full GP) and a Gaussian Predictive process (PPGP) with 64 knots. The oversmoothing by the low-rank predictive process is evident.

Figure 2 serves to reinforce findings that low-rank models may be limited in their ability to produce accurate representation of the underlying process at massive scales. They will need a considerably larger number of basis functions to capture the features of the process and will require substantial computational resources for emulating results from a full GP. As the demands for analyzing large spatial datasets increase from the order of $\sim 10^4$ to $\sim 10^6$ locations, low-rank models may struggle to deliver acceptable inference. In this regard, enhancements such as the multi-resolution predictive process approximations referred to in Section 2.2 are highly promising.

An alternative is to develop full rank models that can exploit sparsity. Instead of deriving basis approximations for w, one could achieve computational gains by modeling either its covariance function or its inverse as sparse. Covariance tapering does the former by modeling $\text{var}\{w\} = K_\theta \odot K_{\text{tap},\nu}$, where $K_{\text{tap},\nu}$ is a sparse covariance matrix formed from a compactly supported, or *tapered*, covariance function with tapering parameter ν and $\odot$ denotes the element wise (or Hadamard) product of two matrices. The Hadamard product of two positive definite matrices is again a positive definite matrix, so $K_\theta \odot K_{\text{tap},\nu}$ is positive definite. Furthermore, $K_{\text{tap},\nu}$ is sparse because a tapered covariance function is equal to 0 for all pairs of locations separated by a distance beyond a threshold ν. We refer the reader to Furrer et al. (2006), Kaufman et al. (2008) and Du et al. (2009) for further computational and theoretical details on covariance tapering. Covariance tapering is undoubtedly an attractive approach for constructing sparse covariance matrices, but its practical implementation for full Bayesian inference will generally require efficient sparse Cholesky decompositions, numerically stable determinant computations and, perhaps most importantly, effective memory management. These issues are yet to be tested for truly massive spatiotemporal datasets with $n \sim 10^5$ or more.

Another way to exploit sparsity is to model the inverse of $\text{var}\{w\}$ as a sparse matrix. For finite-dimensional distributions conditional and simultaneous autoregressive (CAR and SAR) models (see, e.g., Cressie, 1993; Banerjee et al., 2014, and references therein) adopt this approach for areally referenced datasets. More generally, Gaussian Markov

random fields or GMRFs (see, e.g., Rue and Held, 2005) are widely used tools for constructing sparse precision matrices and have led to computational algorithms such as the Integrated Nested Laplace Approximation (INLA) developed by Rue et al. (2009). A subsequent article by Lindgren et al. (2011) show how Gaussian processes can be approximated by GMRFs using computationally efficient sparse representations. Thus, a Gaussian process model with a dense covariance function is approximated by a GMRF with a sparse precision matrix. The approach is very computationally efficient for certain classes of covariance functions generated by a certain class of stochastic partial differential equations (including the versatile Matérn class), but their inferential performance on unobservable spatial, spatiotemporal or multivariate Gaussian processes (perhaps specified through more general covariance or cross-covariance functions) embedded within Bayesian hierarchical models is yet to be assessed.

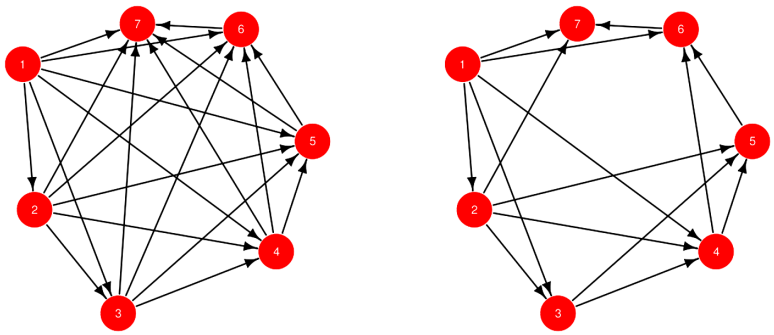
Rather than working with approximations to the process, one could also construct massively scalable sparsity-inducing Gaussian processes that can be conveniently embedded within Bayesian hierarchical models and deliver full Bayesian inference for random fields at arbitrary resolutions. Section 3.1 describes how sparsity is introduced in the precision matrices for graphical Gaussian models by exploiting the relationship between the Cholesky decomposition of a positive definite matrix and conditional independence. These sparse Gaussian models (i.e., normal distributions with sparse precision matrices) can be used prior models for a finite number of spatial random effects. Section 3.2 shows the construction of a process from these graphical Gaussian models. This process will be a Gaussian process whose finite-dimensional realizations will have sparse precision matrices. We call them Nearest Neighbor Gaussian Processes (NNGP). Finally, Section 3.3 outlines how the process can be embedded within hierarchical models and presents some brief simulation examples demonstrating certain aspects of inference from NNGP models.

3.1 Sparse Gaussian graphical models

Consider the hierarchical model (2) and, in particular, the expensive prior density $N(w \mid 0, K_\theta)$. From the dense covariance matrix K_θ, we wish to obtain a covariance matrix $\tilde{K}_\theta$ such that $\tilde{K}_\theta^{-1}$ is sparse and, importantly, its determinant is available cheaply. What would be an effective way of achieving this? One approach would be to consider *modeling* the Cholesky decomposition of the precision matrix so that it is sparse. For example, forcing some elements in the dense half of the triangular Cholesky factor to be zero will introduce sparsity in the precision matrix. To precisely set out which elements should be made zero in the Cholesky factor, we borrow some fundamental notions of sparsity from graphical (Gaussian) models.

The underlying idea is, in fact, ubiquitous in graphical models or Bayesian networks (see, e.g., Lauritzen, 1996; Bishop, 2006; Murphy, 2012). The joint distribution for a random vector w can be looked upon as a directed acyclic graph (DAG) where each node is a random variable w_i. We write the joint distribution as

$$p(w_1)\prod_{i=2}^{n} p(w_i \mid w_1, \ldots, w_{i-1}) = \prod_{i=1}^{n} p(w_i \mid w_{\mathrm{Pa}[i]}) \ ,$$



(a) Full graph (b) Sparse graph

Figure 3: Sparsity using directed acyclic graphs.

where Pa[1] is the empty set and Pa[i] $= \{1, 2, \ldots, i-1\}$ for $i = 2, 3, \ldots, n-1$ is the set of parent nodes with directed edges to i. This model is specific to the ordering (sometimes called "topological ordering") of the nodes. The DAG corresponding to this factorization is shown in Figure 3(a) for $n = 7$ nodes. One can refer to this as the full graphical model since Pa[i] comprises all nodes preceding i in the topological order. Shrinking Pa[i] from the set of all nodes preceding i to a smaller subset of parent nodes yields a different, but still valid, joint distribution. In spatial settings, each of the nodes in the DAG have associated spatial coordinates. Thus, the parents for any node i can be chosen to include a certain fixed number of "nearest neighbors", say based upon their distance from node i. For example, Figure 3(b) shows the DAG when some of the edges are deleted so as to retain at most 3 nearest neighbors in the conditional probabilities. The resulting joint density is

$$p(w_1) \times p(w_2 \,|\, w_1) \times p(w_3 \,|\, w_1, w_2) \times p(w_4 \,|\, w_1, w_2, w_3) \times p(w_5 \,|\, \cancel{w_1}, w_2, w_3, w_4)$$
$$\times\, p(w_6 \,|\, w_1, \cancel{w_2}, \cancel{w_3}, w_4, w_5) \times p(w_7 \,|\, w_1, w_2, \cancel{w_3}, \cancel{w_4}, \cancel{w_5}, w_6)\ .$$

The above model posits that any node i, given its parents, is conditionally independent of any other node that is neither its parent nor its child.

Applying the above notion to multivariate Gaussian densities evinces the connection between conditional independence in DAGs and sparsity. Consider an $n \times 1$ random vector w distributed as $N(0, K_\theta)$. Writing $N(w \,|\, 0, K_\theta)$ as $p(w_1) \prod_{i=2}^{n} p(w_i \,|\, w_1, w_2, \ldots, w_{i-1})$ is equivalent to the following set of linear models,

$$w_1 = 0 + \eta_1 \qquad \text{and} \qquad w_i = a_{i1}w_1 + a_{i2}w_2 + \cdots + a_{i,i-1}w_{i-1} + \eta_i \ \text{ for } i = 2, \ldots, n\ ,$$

or, more compactly, simply $w = Aw + \eta$, where A is $n \times n$ strictly lower-triangular with elements $a_{ij} = 0$ whenever $j \geq i$ and $\eta \sim N(0, D)$ and D is diagonal with diagonal entries $d_{11} = \text{var}\{w_1\}$ and $d_{ii} = \text{var}\{w_i \,|\, w_j : j < i\}$ for $i = 2, \ldots, n$.

From the structure of A it is evident that $I - A$ is nonsingular and $K_\theta = (I - A)^{-1} D (I - A)^{-\top}$. The possibly nonzero elements of A and D are completely determined

by the matrix K_θ. Let `a[i,j]`, `d[i,j]` and `K[i,j]` denote the (i,j)-th entries of A, D and K_θ, respectively. Note that `d[1,1] = K[1,1]` and the first row of A is 0. A pseudocode to compute the remaining elements of A and D is:

```
for(i in 1:(n-1)) {
    a[i+1,1:i] = solve(K[1:i,1:i], K[1:i,i+1])
    d[i+1,i+1] = K[i+1,i+1] - dot(K[i+1,1:i],a[i+1,1:i])
}.
```
(16)

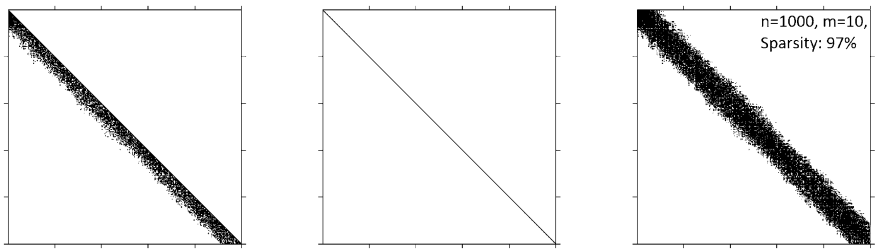
Here `a[i+1,1:i]` is the $1 \times$ `i` row vector comprising the possibly nonzero elements of the `i+1`-th row of A, `K[1:i,1:i]` is the `i` $\times$ `i` leading principal submatrix of K_θ, `K[1:i, i]` is the `i` $\times 1$ row vector formed by the first `i` elements in the `i`-th column of K_θ, `K[i, 1:i]` is the $1 \times$ `i` row vector formed by the first `i` elements in the `i`-th row of K_θ, `solve(B,b)` computes the solution for the linear system `Bx = b`, and `dot(u,v)` provides the inner product between vectors `u` and `v`. The determinant of K_θ is obtained with almost no additional cost: it is simply $\prod_{\mathtt{i}=1}^{\mathtt{n}} \mathtt{d[i,i]}$.

The above pseudocode provides a way to obtain the Cholesky decomposition of K_θ. If $K_\theta = LDL^\top$ is the Cholesky decomposition, then $L = (I - A)^{-1}$. There is, however, no apparent gain to be had from the preceding computations since one will need to solve increasingly larger linear systems as the loop runs into higher values of `i`. Nevertheless, it immediately shows how to exploit sparsity if we set some of the elements in the lower triangular part of A to be zero. For example, suppose we set at most `m` elements in each row of A to be nonzero. Let `N[i]` be the set of indices `j` $<$ `i` such that `a[i,j]` $\neq 0$. We can compute the nonzero elements of A and the diagonal elements of D efficiently as:

```
for(i in 1:(n-1) {
    Pa = N[i+1] # neighbors of i+1
    a[i+1,Pa] = solve(K[Pa,Pa], K[(i+1),Pa])
    d[i+1,i+1] = K[i+1,i+1] - dot(K[(i+1),Pa], a[i+1,Pa])
}.
```
(17)

In (17) we solve `n-1` linear systems of size at most `m` $\times$ `m`. This can be performed in $\sim \mathtt{nm}^3$ flops, whereas the earlier pseudocode in (16) for the dense model required $\sim \mathtt{n}^3$ flops. These computations can be performed in parallel as each iteration of the loop is independent of the others.

The above discussion provides a very useful strategy for introducing sparsity in a precision matrix. Let K_θ and K_θ^{-1} both be dense $n \times n$ positive definite matrices. Suppose we use the pseudocode in (17) with `K` $= K_\theta$ to construct a sparse strictly lower-triangular matrix A with no more than m non-zero entries in each row, where m is considerably smaller than n, and the diagonal matrix D. The resulting matrix $\tilde{K}_\theta = (I-A)^{-1}D(I-A)^{-\top}$ is a covariance matrix whose inverse $\tilde{K}_\theta^{-1} = (I - A^\top)D^{-1}(I-A)$ is sparse. Figure 4 presents a visual representation of the sparsity. While $\tilde{K}_\theta$ need not be sparse, the density $N(w \,|\, 0, \tilde{K}_\theta)$ is cheap to compute since $\tilde{K}_\theta^{-1}$ is sparse and $\det(\tilde{K}_\theta^{-1})$ is the product of the diagonal elements of D^{-1}. Therefore, one way to achieve massive scalability for models such as (2) is to assume that w has prior $N(w \,|\, 0, \tilde{K}_\theta)$ instead of $N(w \,|\, 0, K_\theta)$.



(a) $I - A$ (b) D^{-1} (c) $\tilde{K}_\theta^{-1}$

Figure 4: Structure of the factors making up the sparse $\tilde{K}_\theta^{-1}$ matrix.

3.2 From distributions to processes

If we are interested in estimating the spatial or spatiotemporal process parameters from a finite collection of random variables, then we can use the approach in Section 3.1 with $w_i := w(\ell_i)$. In spatial settings, matters are especially convenient as we can delete the edges in the DAG based upon the distances among ℓ_i's. In fact, one can decide to retain at most m of the nearest neighbors for each location and delete all remaining edges. This implies that the (i, j)-th element of A in Section 3.1 will be nonzero only if ℓ_j is one of the m nearest neighbors of ℓ_i. In fact, this idea has been effectively used to construct composite likelihoods for Gaussian process models by Vecchia (1988) and Stein et al. (2004), while Stroud et al. (2017) exploits this idea to propose preconditioned conjugate gradient algorithms for Bayesian and maximum likelihood estimates on large incomplete lattices.

Localized Gaussian process regression based on few nearest neighbors has also been used to obtain fast kriging estimates. Emery (2009) provides fast updates for kriging equations after adding a new location to the input set. Iterative application of their algorithm yields a localized kriging estimate based on a small set of locations (including few nearest neighbors). The local estimate often provides an excellent approximation to the global kriging estimate which uses data observed at all the locations to predict at a new location. However, this assumes that the parameters associated with the mean and covariance of the GP are known or already estimated. Local Approximation GP, or LAGP (Gramacy and Apley, 2015; Gramacy and Haaland, 2016; Gramacy, 2016), extends this further to estimate the parameters at each new location, essentially providing a non-stationary local approximation to a Gaussian Process at every predictive location and can be used to interpolate or smooth the observed data.

If, however, posterior predictive inference is sought at arbitrary spatiotemporal resolutions, i.e., for the entire process $\{w(\ell) : \ell \in \mathcal{L}\}$, then the ideas in Section 3.1 need to be extended to process-based models. Recently, Datta et al. (2016a) proposed a Nearest Neighbor Gaussian Process (NNGP) for modeling large spatial data. NNGP is a well defined Gaussian Process over a domain $\mathcal{L}$ and yields finite dimensional Gaussian densities with sparse precision matrices. This has been also extended to a dynamic NNGP with dynamic neighbor selection for massive spatiotemporal data (Datta et al., 2016b). The

NNGP delivers massive scalability both in terms of parameter estimation and kriging. Unlike low rank processes, it does not oversmooth and accurately emulates the inference from full rank GPs.

We will construct the NNGP in two steps. First, we specify a multivariate Gaussian distribution over a fixed finite set r points in $\mathcal{L}$, say $\mathcal{R} = \{\ell_1^*, \ell_2^*, \ldots, \ell_r^*\}$, which we call the *reference set*. The reference set can be very large. It can be a fine grid of points over $\mathcal{L}$ or one can simply take $r = n$ and let $\mathcal{R}$ be the set of observed points in $\mathcal{L}$. We require that the inverse of the covariance matrix be sparse and computationally efficient. Therefore, we specify that $w_{\mathcal{R}} \sim N(0, \tilde{K}_\theta)$, where $w_{\mathcal{R}}$ is the $r \times 1$ vector with elements $w(\ell_i^*)$ and $\tilde{K}_\theta$ is a covariance matrix such that $\tilde{K}_\theta^{-1}$ is sparse. The matrix $\tilde{K}_\theta$ is constructed from a dense covariance matrix K_θ as described in Section 3.1. This provides a highly effective approximation (Vecchia, 1988; Stein et al., 2004) as below:

$$N(w_{\mathcal{R}} \,|\, 0, K_\theta) = \prod_{i=1}^{r} p(w(\ell_i^*) \,|\, w_{H(\ell_i^*)}) \approx \prod_{i=1}^{r} p(w(\ell_i^*) \,|\, w_{N(\ell_i^*)}) = N(w_{\mathcal{R}} \,|\, 0, \tilde{K}_\theta) \,, \quad (18)$$

where *history sets* $H(\ell_i^*)$ so that $H(\ell_1^*)$ is the empty set and $H(\ell_i^*) = \{\ell_1^*, \ell_2^*, \ldots, \ell_{i-1}^*\}$ for $i = 2, 3, \ldots, r$ and we have much smaller *neighbor sets* $N(\ell_i^*) \subseteq H(\ell_i^*)$ for each ℓ_i^* in $\mathcal{R}$. We have legitimate probability models for any choice of $N(\ell_i^*)$'s as long as $N(\ell_i^*) \subseteq H(\ell_i^*)$. One easy specification is to define $N(\ell_i^*)$ as the set of m nearest neighbors of ℓ_i^* among the points in $\mathcal{R}$. Therefore,

$$N(\ell_i) = \begin{cases} \text{empty set for } i = 1 \\ H(\ell_i^*) = \{\ell_1^*, \ell_2^*, \ldots, \ell_{i-1}^*\} \;\text{ for }\; i = 2, 3, \ldots, m \\ m \text{ nearest neighbors of } \ell_i^* \text{ among } H(\ell_i^*) \;\text{ for } i = m+1, \ldots, n \end{cases} .$$

If $m(<< r)$ denotes the limiting size of the neighbor sets $N(\ell)$, then $\tilde{K}_\theta^{-1}$ has at most $O(rm^2)$ non-zero elements. Hence, the approximation in (18) produces a sparsity-inducing proper prior distribution for random effects over $\mathcal{R}$ that closely approximates the realizations from a $GP(0, K_\theta)$.

To construct the NNGP we extend the above model to arbitrary locations. We define neighbor sets $N(\ell)$ for any $\ell \in \mathcal{L}$ as the set of m nearest neighbors of ℓ in $\mathcal{R}$. Thus, $N(\ell) \subseteq \mathcal{R}$ and the process can be derived from $p(w_{\mathcal{R}}, w(\ell) \,|\, \theta) = N(w_{\mathcal{R}} \,|\, 0, \tilde{K}_\theta) \times p\left(w(\ell) \,|\, w_{N(\ell)}, \theta\right)$ or, equivalently, by writing

$$w(\ell) = \sum_{i=1}^{r} a_i(\ell) w(\ell_i^*) + \eta(\ell) \text{ for any } \ell \notin \mathcal{R} \,, \quad (19)$$

where $a_i(\ell) = 0$ whenever $\ell_i^* \notin N(\ell)$, $\eta(\ell) \overset{ind}{\sim} N(0, \delta^2(\ell))$ is a process independent of $w(\ell)$, $\text{Cov}\{\eta(\ell), \eta(\ell')\} = 0$ for any two distinct points in $\mathcal{L}$, and

$$\delta^2(\ell) = K_\theta(\ell, \ell) - K_\theta(\ell, N(\ell)) K_\theta^{-1}(N(\ell), N(\ell)) K_\theta(N(\ell), \ell) \,.$$

Taking conditional expectations in (19) yields $\text{E}[w(\ell) \,|\, w_{N(\ell)}] = \sum_{i:\ell_i \in N(\ell)} a_i(\ell) w(\ell_i^*)$, which implies that for each ℓ the nonzero $a_i(\ell)$'s are obtained by solving an $m \times m$ linear system. The above construction ensures that $w(\ell)$ is a legitimate Gaussian process

whose realizations over any finite collection of arbitrary points in $\mathcal{L}$ will have a multivariate normal distribution with a sparse precision matrix. More formal developments and technical details in the spatial and spatiotemporal settings can be found in Datta et al. (2016a) and Datta et al. (2016b), respectively.

One point worth considering is the definition of "neighbors." There is some flexibility here. In the spatial setting, the correlation functions usually decay with increasing inter-site distance, so the set of nearest neighbors based on the inter-site distances represents locations exhibiting highest correlation with the given locations. For example, on the plane one could simply use the Euclidean metric to construct neighbor sets, although Stein et al. (2004) recommend including a few points that are farther apart. The neighbor sets can be fixed before the model fitting exercise.

In spatiotemporal settings, matters are more complicated. Spatiotemporal covariances between two points typically depend on the spatial as well as the temporal lag between the points. Non-separable isotropic spatiotemporal covariance functions can be written as $K_\theta((s_1,t_1),(s_2,t_2)) = K_\theta(h,u)$ where $h = \|s_1 - s_2\|$ and $u = |t_1 - t_2|$. This often precludes defining any universal distance function $d : (\mathcal{S}\times\mathcal{T})^2 \to \Re^+$ such that $K_\theta((s_1,t_1),(s_2,t_2))$ will be monotonic with respect to $d((s_1,t_1),(s_2,t_2))$ for all choices of θ. This makes it difficult to define universal nearest neighbors in spatiotemporal domains. To obviate this hurdle, Datta et al. (2016b) define "nearest neighbors" in a spatiotemporal domain using the spatiotemporal covariance function itself as a proxy for distance. This can work for arbitrary domains. For any three points ℓ_1, ℓ_2 and ℓ_3, we say that ℓ_1 is nearer to ℓ_2 than to ℓ_3 if $K_\theta(\ell_1,\ell_2) > K_\theta(\ell_1,\ell_3)$. Subsequently, this definition of "distance" is used to find m nearest neighbors for any location.

However, for every point ℓ_i, its neighbor set $N_\theta(\ell)$ will now depend on θ and can change from iteration to iteration in the estimation algorithm. If θ were known, one could have simply evaluated the pairwise correlations between any point ℓ_i^* in $\mathcal{R}$ and all points in its history set $H(\ell_i^*)$ to obtain $N_\theta(\ell_i^*)$ – the set of m true nearest neighbors. In practice, however, θ is unknown and for every new value of θ in an iterative algorithm, we need to search for the neighbor sets within the history sets. Since the history sets are quite large, searching the entire space for nearest neighbors in each iteration will be computationally unfeasible. Datta et al. (2016b) offer some smart strategies for selecting spatiotemporal neighbors. They propose restricting the search for the neighbor sets to carefully constructed small subsets of the history sets. These small *eligible sets* $E(\ell_i^*)$ are constructed in such a manner that, despite being much smaller than the history sets, they are guaranteed to contain the true nearest neighbor sets. This strategy works when we choose m to be a perfect square and the original nonseparable covariance function $K_\theta(h,u)$ satisfies *natural monotonicity*, i.e. $K_\theta(h,u)$ is decreasing in h for fixed u and decreasing in u for fixed h. All Matèrn-based space-time separable covariances and many non-separable classes of covariance functions possess this property (Stein, 2013; Omidi and Mohammadzadeh, 2015).

3.3 Hierarchical NNGP models

We briefly turn to model fitting and estimation. For the approximation in (18) to be effective, the size of the reference set, r, needs to be large enough to represent the spatial

domain. However, this does not impede computations involving NNGP models because the storage and number of floating point operations are always linear in r. The reference set $\mathcal{R}$ can, in principle, be any finite set of locations in the study domain. A particularly convenient choice, in practice, is to simply take $\mathcal{R}$ to be the set of observed locations in the dataset. Datta et al. (2016a) demonstrate through extensive simulation experiments and a real application that this simple choice seems to be very effective.

Since the NNGP is a proper Gaussian process, we can use it as a prior for the spatial random effects in any hierarchical model. We write $w(\ell) \sim NNGP(0, \tilde{K}_\theta(\cdot,\cdot))$, where $\tilde{K}_\theta(\ell,\ell')$ is the covariance function for the NNGP (see Datta et al., 2016a, for a closed form expression). For example, with $r = n$ and $\mathcal{R}$ the set of observed locations, one can build a scalable Bayesian hierarchical model exactly as with a usual spatial process, but assigning an NNGP to the spatial random effects. Here is a simple NNGP-based spatial model with a first stage exponential family model:

$$\begin{array}{rl} Y(\ell)\,|\,g(\cdot),\beta,w(\ell) & \overset{ind}{\sim} P_\tau \quad \text{exponential family}\,, \\ g(\mathrm{E}[Y(\ell)]) & = x^\top(\ell)\beta + w(\ell)\,, \quad w(\ell) \sim NNGP(0,\tilde{K}_\theta(\cdot,\cdot))\,, \\ \{\theta,\beta,\tau\} & \sim p(\theta,\beta,\tau)\,, \end{array} \tag{20}$$

where P_τ is an exponential family distribution with link function $g(\cdot)$. Posterior sampling from (20) is customarily performed using Gibbs sampling with Metropolis steps. Computational benefits emerge from the fact that the full conditional distribution $p(w(\ell_i)\,|\,w_{\mathcal{R}},\theta,\beta,\tau) = p(w(\ell_i)\,|\,w_{N(\ell_i)},\theta,\beta,\tau)$ and since $w_{N(\ell_i)}$ is an $m\times 1$ subset of $w_{\mathcal{R}}$. Prediction at any arbitrary location $\ell \notin \mathcal{R}$ is performed by sampling from the posterior predictive distribution. For each draw of $\{w_{\mathcal{R}},\beta,\theta,\tau\}$ from $p(w_{\mathcal{R}},\beta,\tau,\theta\,|\,y)$, we draw a $w(\ell)$ from $N(a^\top(\ell)w_{N(\ell)},\delta^2(\ell))$ and $y(\ell)$ from $p(y(\ell)\,|\,\beta,w(\ell),\tau)$, where y is the vector of observed outcomes and $a(\ell)$ is a vector of the nonzero $a_j(\ell)$'s in (19).

Another, even simpler, example could be modeling a continuous outcome itself as an NNGP. Let the desired full GP specification be $Y(\ell) \sim GP(x^\top(\ell)\beta, K_\theta(\cdot,\cdot))$. We derive the NNGP from this K_θ and obtain

$$Y(\ell) \sim NNGP(\mu(\ell),\tilde{K}_\theta(\cdot,\cdot))\,; \quad \mu(\ell) = x^\top(\ell)\beta\,; \quad \{\theta,\beta\} \sim p(\theta,\beta)\,. \tag{21}$$

The above model is extremely fast. The likelihood is of the form $y \sim N(X\beta,\tilde{K}_\theta)$, where $\tilde{K}_\theta^{-1} = (I-A^\top)D^{-1}(I-A)$ is sparse and A and D are obtained from (17) efficiently in parallel. The parameter space of interest is $\{\theta,\beta\}$, which is much smaller than for (20) where the latent spatial process also was unknown. While (21) does not separate the residuals into a spatial process and a measurement error process, one can still include measurement error variance, or the nugget, in (21). Here, one would absorb the nugget into θ. For example, suppose we wish to approximate (1) using (21). We could write the likelihood in (1) as $N(y\,|\,X\beta,K_\theta)$, where $K_\theta = \sigma^2 R_\phi + \tau^2 I_n$, R_ϕ is a spatial correlation matrix and $\theta = \{\sigma^2,\phi,\tau^2\}$. These will also feature in the derived NNGP covariance matrix $\tilde{K}_\theta$. We can predict the outcome at an arbitrary point ℓ by sampling from the posterior predictive distribution as follows: for each draw of $\{\beta,\theta\}$ from $p(\beta,\theta\,|\,y)$, we draw a $y(\ell)$ from $N(y(\ell)\,|\,x^\top(\ell)\beta,\delta^2(\ell))$. Note, however, that there is no latent smooth process $w(\ell)$ in (21) and inference on the latent spatial process is precluded.

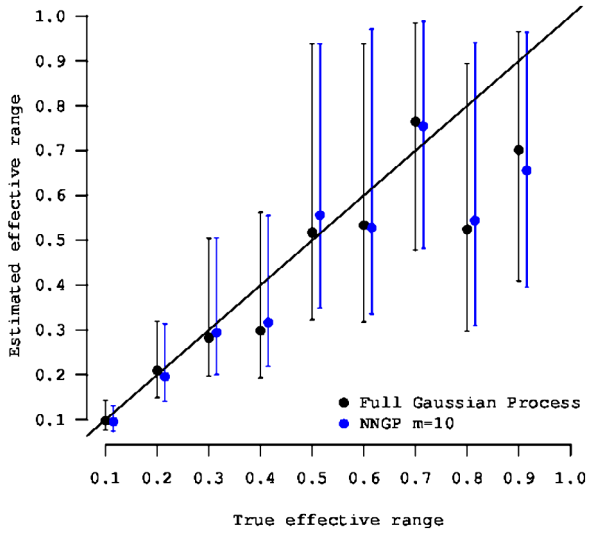


Figure 5: 95% credible intervals for the effective spatial range from an NNGP model with $m = 10$ and a full GP model fitted to 10 different simulated datasets with true effective range fixed at values between 0.1 and 1.0 in increments of 0.1.

Likelihood computations in NNGP models usually involve $O(nm^3)$ flops. One does not need to store $n \times n$ matrices, only $m \times m$ matrices which leads to storage $\sim nm^2$. Substantial computational savings accrue because m is usually very small. Datta et al. (2016a) demonstrate that fitting NNGP models to the simulated data in Figure 2 with number of neighbors as less as $m = 10$ produce posterior estimates of the spatial surface indistinguishable from Figures 2(a) and 2(b). In fact, simulation experiments in Datta et al. (2016a) and Datta et al. (2016b) also affirm that m can usually be taken to be very small compared to r; there seems to be no inferential advantage to taking m to exceed 15, even for datasets with over 10^5 spatial locations. For example, Figure 5 shows the 95% posterior credible intervals for a series of 10 simulation experiments where the true effective range was fixed at values from 0.1 to 1.0 in increments of 0.1. Each dataset comprised 2500 points. Even with $m = 10$ neighbors, the credible intervals for the effective spatial range from the NNGP model were very consistent with those from the full GP model. Datta et al. (2016a) present simulations using the Matérn and other covariance functions revealing very similar behavior.

Another important point to note is that $\tilde{K}_\theta$ is not invariant to the order in which we define $H(\ell_1) \subseteq H(\ell_2) \subseteq \cdots \subseteq H(\ell_r)$ (i.e., the topological order). Vecchia (1988) and Stein et al. (2004) both assert that the approximation in (18) is not sensitive to this ordering. This is corroborated by simulation experiments by Datta et al. (2016a), but a recent manuscript by Guinness (2016) has indicated sensitivity to the ordering in terms of model deviance. We conducted some preliminary investigations to investigate the effect of the topological order. In one simple experiment we generated data from the "true" model in (1) for 6400 spatial locations arranged over an 80×80 grid. The

NNGP from different topological orders					
	True	Sorted coord(x+y)	MMD	Sorted x	Sorted y
σ	1	0.79 (0.69, 1.04)	0.80 (0.69, 1.02)	0.80 (0.70, 1.05)	0.83 (0.69, 1.08)
τ	0.45	0.45 (0.44, 0.46)	0.45 (0.44, 0.47)	0.45 (0.44, 0.46)	0.45 (0.44, 0.47)
ϕ	5	8.11 (4.42, 11.10)	7.63 (4.58, 10.97)	8.01 (4.26, 11.18)	7.12 (4.06, 11.03)
KL-D	–	24.04022	13.88847	22.30667	21.59174
RMSPE	–	0.5278996	0.5278198	0.527912	0.527807

Table 2: Posterior parameter estimates, the Kullback–Leibler divergence (KL-D) and root mean square predictive errors (RMSPE) are presented for four NNGP models constructed from different topological orderings. The four orderings from left to right are "sorted on the sum of vertical and horizontal coordinate", maximum-minimum distance (Guinness, 2016), sorted on horizontal coordinate and sorted on vertical coordinate.

parameter β in (1) was set to 0, the covariance function was specified as $K_\theta(\ell_i, \ell_j) = \sigma^2 \exp(-\phi\|\ell_i - \ell_j\|)$, and $\epsilon(\ell_i) \overset{iid}{\sim} N(0, \tau^2)$ with the true values of σ^2, ϕ and τ^2 given in the second column of Table 2. Four different NNGP models corresponding to (21) with $\tilde{K}_\theta$ derived from $K_\theta = \sigma^2 R_\phi + \tau^2 I$ and R_ϕ having elements $\exp(-\phi\|\ell_i - \ell_j\|)$, were fitted to the simulated data. Each of these models were constructed with $m = 10$ nearest neighbors, but with different ordering of the points $\ell = (x, y)$. These were performed according to the sum of the coordinates $x + y$, a maximum-minimum distance (MMD) proposed by Guinness (2016), the x coordinate, and the y coordinate. Table 2 presents a comparison of these NNGP models. Irrespective of the ordering of the points, the inference with respect to parameter estimates and predictive performance is extremely robust and effectively indistinguishable from each other. However, the posterior mean of the Kullback–Leibler divergence of these models from the true generating model revealed that the metric proposed by Guinness (2016) is indeed less than the other three. Further explorations are currently being conducted to see how this behavior changes for more complex nonstationary models and in more general settings.

4 Discussion and future directions

The article has attempted to provide some insight into constructing highly scalable Bayesian hierarchical models for very large spatiotemporal datasets using low-rank and sparsity-inducing processes. Such models are increasingly being employed to answer complex scientific questions and analyze massive spatiotemporal datasets in the natural and environmental sciences. Any standard Bayesian estimation algorithm, such as Markov chain and Hamiltonian Monte Carlo (see, e.g., Robert and Casella, 2004; Brooks et al., 2011; Gelman et al., 2013; Neal, 2011; Hoffman and Gelman, 2014), Integrated Nested Laplace Approximations (Rue et al., 2009), and Variational Bayes (see, e.g., Bishop, 2006) can be used for fitting these models. The models ensure that the algorithmic complexity has $\sim n$ floating point operations (flops), where n the number of spatial locations (per iteration). Storage requirements are also linear in n. Methods

such as the multiresolution predictive process (Katzfuss, 2017) and the NNGP (Datta et al., 2016a) can scale up to datasets in the order of $\sim 10^6$ spatial and/or temporal points without sacrificing richness in the model.

While the NNGP certainly seem to have an edge in scalability over the more conventional low-rank or fixed rank models, it is premature to say whether its inferential performance will always excel over low rank of fixed rank models. For example, analyzing complex nonstationary random fields may pose challenges regarding construction of neighbor sets as simple distance-based definition of neighbors may prove to be inadequate. Multiresolution basis functions may be more adept at capturing nonstationary, but may struggle with massive datasets. Dynamic neighbor selection for nonstationary fields, where neighbors will be chosen based upon the covariance kernel itself, analogous to Datta et al. (2016b) for space-time covariance functions, may be an option worth exploring. Multiresolution NNGPs, where the residual from the NNGP approximation is modeled hierarchically (analogous to Katzfuss, 2017, for the predictive process) may also be promising in terms of full Bayesian inference at massive scales.

There remain other challenges in high-dimensional geostatistics. Here, we have considered geostatistical settings where we have very large numbers of locations and/or time-points, but restricted our discussion to univariate outcomes. In practice, we often observe a $q \times 1$ variate response $y(\ell)$ along with a set of explanatory variables $X(\ell)$ and $q \times 1$ variate GP, $w(\ell)$, is used to capture the spatial patterns beyond the observed covariates. We seek to capture associations among the variables as well as the strength of spatiotemporal association for each outcome. One specific geostatistical problem in ecology that currently lacks a satisfying solution is a joint species distribution model, where we seek to model a large collection of species (say, order 10^3) over a large collection of spatial sites (again, say, order 10^3).

The linear model of coregionalization (LMC) proposed by Matheron (1982) is among the most general models for multivariate spatial data analysis. Here, the spatial behavior of the outcomes is assumed to arise from a linear combination of the independent latent processes operating at different spatial scales (Chilés and Delfiner, 1999). The idea resembles latent factor analysis (FA) models for multivariate data analysis (e.g., Anderson, 2003) except that in the LMC the number of latent processes is usually taken to be the same as the number of outcomes. Then, an $q \times q$ covariance matrix has to be estimated for each spatial scale (see, e.g., Lark and Papritz, 2003; Castrignanó et al., 2005; Zhang, 2007), where q is the number of outcomes. When q is large (e.g., $q \geq 5$ and 300 spatial locations), obtaining such estimates is expensive. Schmidt and Gelfand (2003) and Gelfand et al. (2004) associate only a $q \times q$ triangular matrix with the latent processes. However, high dimensional outcomes are still computationally prohibitive for these models.

Spatial factor models (see, e.g., Lopes and West, 2004; Lopes et al., 2008; Wang and Wall, 2003) have been used to handle high dimensional outcomes but with modest number of spatial locations. Dimension reduction is needed in two aspects: (i) the length of the vector of outcomes, and (ii) the very large number of spatial locations. Latent variable (factor) models are usually used to address the former, while low-rank spatial processes offer a rich and flexible modeling option for dealing with a large number of

locations. Ren and Banerjee (2013) have exploited these two ideas to propose a class of hierarchical low-rank spatial factor models and also explored stochastic selection of the latent factors without resorting to complex computational strategies (such as reversible jump algorithms) by utilizing certain identifiability characterizations for the spatial factor model. Their model was designed to capture associations among the variables as well as the strength of spatial association for each variable. In addition, they reckoned with the common setting where not all the variables have been observed over all locations, which leads to *spatial misalignment*. The fully Bayesian approach effectively deals with spatial misalignment. However, this method is likely to suffer from the limited ability of low-rank models to scale to a very large number of locations. Promising ideas include using the multiresolution predictive process or the NNGP as a prior on the spatial factors.

Computational developments with regard to Markov chain Monte Carlo (MCMC) algorithms (see, e.g., Robert and Casella, 2004) have contributed enormously to the dissemination of Bayesian hierarchical models in a wide array of disciplines. Spatial modeling is no exception. However, the challenges for automated implementation of geostatistical model fitting and inference are substantial. First, expensive matrix computations are required that can become prohibitive with large datasets. Second, routines to fit unmarginalized models are less suited for direct updating using a Gibbs sampler and result in slower convergence of the chains. Third, investigators often encounter multivariate spatial datasets with several spatially dependent outcomes, whose analysis requires multivariate spatial models that involve demanding matrix computations. These issues have, however, started to wane with the delivery of relatively simpler software packages in the `R` statistical computing environment via the Comprehensive R Archive Network (CRAN) (`http://cran.r-project.org`). Several packages that automate Bayesian methods for point-referenced data and diagnose convergence of MCMC algorithms are easily available from CRAN. Packages that fit Bayesian models include `geoR`, `geoRglm`, `spTimer`, `spBayes`, `spate`, and `ramps`.

In terms of the hierarchical geostatistical models presented in this article, `spBayes` offers users a suite of Bayesian hierarchical models for Gaussian and non-Gaussian univariate and multivariate spatial data as well as dynamic Bayesian spatio-temporal models. It focuses upon performance issues for full Bayesian inference, sampler convergence rate and efficiency using a collapsed Gibbs sampler, decreasing sampler run-time by avoiding expensive matrix computations, and increased scalability to large datasets by implementing predictive process models. Beyond these general computational improvements for existing models, it analyzes data indexed both in space and time using a class of dynamic spatiotemporal models, and their predictive process counterparts, for settings where space is viewed as continuous and time is taken as discrete. Finally, we have modeling environments such as `Nimble` (de Valpine et al., 2017) that gives users enormous flexibility to choose algorithms for fitting their models, and `Stan` (Carpenter et al., 2017) that estimates Bayesian hierarchical models using Hamiltonian dynamics. The NNGP and the predictive process can be also coded in `Nimble` and `Stan` fairly easily.

References

Anderson, T. W. (2003). *An Introduction to Multivariate Statistical Analysis*. New York, NY: Wiley, third edition. MR1990662. 606

Banerjee, S., Carlin, B. P., and Gelfand, A. E. (2014). *Hierarchical Modeling and Analysis for Spatial Data*. Boca Raton, FL: Chapman & Hall/CRC, second edition. MR3362184. 583, 584, 588, 591, 596

Banerjee, S., Finley, A. O., Waldmann, P., and Ericcson, T. (2010). "Hierarchical spatial process models for multiple traits in large genetic trials." *Journal of the American Statistical Association*, 105: 506–521. MR2724841. doi: http://dx.doi.org/10.1198/jasa.2009.ap09068. 592

Banerjee, S., Gelfand, A. E., Finley, A. O., and Sang, H. (2008). "Gaussian predictive process models for large spatial datasets." *Journal of the Royal Statistical Society, Series B*, 70: 825–848. 586, 590, 591

Barry, R. and Ver Hoef, J. (1996). "Blackbox kriging: Spatial prediction without specifying variogram models." *Journal of Agricultural, Biological and Environmental Statistics*, 1: 297–322. 587

Bevilacqua, M. and Gaetan, C. (2014). "Comparing composite likelihood methods based on pairs for spatial Gaussian random fields." *Statistics and Computing*, 1–16. 586

Bishop, C. (2006). *Pattern Recognition and Machine Learning*. New York, NY: Springer-Verlag. MR2247587. doi: http://dx.doi.org/10.1007/978-0-387-45528-0. 597, 605

Brooks, S., Gelman, A., Jones, G. L., and Meng, X.-L. (2011). *Handbook of Markov Chain Monte Carlo*. Boca Raton, FL: CRC Press. 605

Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., and Riddell, A. (2017). "Stan: A probabilistic programming language." *Journal of Statistical Software*, 76(1): 1–32. https://www.jstatsoft.org/index.php/jss/article/view/v076i01. 607

Castrignanó, A., Cherubini, C., Giasi, C., Castore, M., Mucci, G. D., and Molinari, M. (2005). "Using Multivariate Geostatistics for Describing Spatial Relationships among some Soil Properties." In *ISTRO Conference Brno*. 606

Chilés, J. and Delfiner, P. (1999). *Geostatistics: Modeling Spatial Uncertainty*. John Wiley: New York. 606

Crainiceanu, C. M., Diggle, P. J., and Rowlingson, B. (2008). "Bivariate binomial spatial modeling of Loa Loa prevalence in tropical Africa." *Journal of the American Statistical Association*, 103: 21–37. MR2420211. doi: http://dx.doi.org/10.1198/016214507000001409. 586

Cressie, N. (1993). *Statistics for Spatial Data*. Wiley-Interscience, revised edition. MR1239641. doi: http://dx.doi.org/10.1002/9781119115151. 583, 584, 596

Cressie, N. and Johannesson, G. (2008). "Fixed rank kriging for very large data sets." *Journal of the Royal Statistical Society, Series B*, 70: 209–226. 586, 587

Cressie, N., Shi, T., and Kang, E. L. (2010). "Fixed rank filtering for spatio-temporal data." *Journal of Computational and Graphical Statistics*, 19: 724–745. MR2732500. doi: http://dx.doi.org/10.1198/jcgs.2010.09051. 586, 587

Cressie, N. A. C. and Wikle, C. K. (2011). *Statistics for Spatio-temporal Data*. Wiley series in probability and statistics. Hoboken, N.J. Wiley. http://opac.inria.fr/record=b1133266 MR2848400. 583, 584

Datta, A., Banerjee, S., Finley, A. O., and Gelfand, A. E. (2016a). "Hierarchical nearest-neighbor Gaussian process models for large geostatistical datasets." *Journal of the American Statistical Association*, 111: 800–812. doi: http://dx.doi.org/10.1080/01621459.2015.1044091. 595, 600, 602, 603, 604, 606

Datta, A., Banerjee, S., Finley, A. O., Hamm, N. A. S., and Schaap, M. (2016b). "Non-separable dynamic nearest-neighbor Gaussian process models for large spatio-temporal data with an application to particulate matter analysis." *Annals of Applied Statistics*, 10: 1286–1316. MR3553225. doi: http://dx.doi.org/10.1214/16-AOAS931. 600, 602, 604, 606

de Valpine, P., Turek, D., Paciorek, C., Anderson-Bergman, C., Temple Lang, D., and Bodik, R. (2017). "Programming with models: Writing statistical algorithms for general model structures with NIMBLE." *Journal of Computational and Graphical Statistics*, 26: 403–413. doi: http://dx.doi.org/10.1080/10618600.2016.1172487. 607

Du, J., Zhang, H., and Mandrekar, V. S. (2009). "Fixed-domain asymptotic properties of tapered maximum likelihood estimators." *Annals of Statistics*, 37: 3330–3361. MR2549562. doi: http://dx.doi.org/10.1214/08-AOS676. 586, 596

Eidsvik, J., Shaby, B. A., Reich, B. J., Wheeler, M., and Niemi, J. (2014). "Estimation and prediction in spatial models with block composite likelihoods." *Journal of Computational and Graphical Statistics*, 23: 295–315. 586

Emery, X. (2009). "The kriging update equations and their application to the selection of neighboring data." *Computational Geosciences*, 13(3): 269–280. http://dx.doi.org/10.1007/s10596-008-9116-8. 600

Finley, A. O., Banerjee, S., and Gelfand, A. E. (2015). "spBayes for large univariate and multivariate point-referenced spatio-temporal data models." *Journal of Statistical Software*, 63(13): 1–28. http://www.jstatsoft.org/v63/i13/. 595

Finley, A. O., Banerjee, S., and McRoberts, R. E. (2009a). "Hierarchical spatial models for predicting tree species assemblages across large domains." *Annals of Applied Statistics*, 3(3): 1052–1079. MR2750386. doi: http://dx.doi.org/10.1214/09-AOAS250. 586

Finley, A. O., Sang, H., Banerjee, S., and Gelfand, A. E. (2009b). "Improving the performance of predictive process modeling for large datasets." *Computational Statistics and Data Analysis*, 53(8): 2873–2884. 592

Fuentes, M. (2007). "Approximate likelihood for large irregularly spaced spatial data." *Journal of the American Statistical Association*, 102(477): 321–331. http://dx.doi.org/10.1198/016214506000000852. 585

Furrer, R., Genton, M. G., and Nychka, D. (2006). "Covariance tapering for interpolation of large spatial datasets." *Journal of Computational and Graphical Statistics*, 15: 503–523. 586, 592, 596

Gelfand, A., Diggle, P., Fuentes, M., and Guttorp, P. (2010). *Handbook of Spatial Statistics*. Boca Raton, FL: CRC Press. 583, 584

Gelfand, A. E., Schmidt, A. M., Banerjee, S., and Sirmans, C. F. (2004). "Nonstationary multivariate process modeling through spatially varying coregionalization." *TEST*, 13(2): 263–312. 606

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013). *Bayesian Data Analysis, 3rd Edition*. Chapman & Hall/CRC Texts in Statistical Science. Chapman & Hall/CRC. 588, 595, 605

Gneiting, T. and Guttorp, P. (2010). "Continuous-parameter Spatio-temporal Processes." In Gelfand, A., Diggle, P., Fuentes, M., and Guttorp, P. (eds.), *Handbook of Spatial Statistics*, 427–436. 584

Gramacy, R. (2016). "laGP: Large-scale spatial modeling via local approximate Gaussian processes in R." *Journal of Statistical Software*, 72(1): 1–46. https://www.jstatsoft.org/index.php/jss/article/view/v072i01 600

Gramacy, R. B. and Apley, D. W. (2015). "Local Gaussian process approximation for large computer experiments." *Journal of Computational and Graphical Statistics*, 24(2): 561–578. http://dx.doi.org/10.1080/10618600.2014.914442. 600

Gramacy, R. B. and Haaland, B. (2016). "Speeding up neighborhood search in local Gaussian process prediction." *Technometrics*, 58(3): 294–303. http://dx.doi.org/10.1080/00401706.2015.1027067. 600

Gramacy, R. B. and Lee, H. K. H. (2008). "Bayesian treed Gaussian process models with an application to computer modeling." *Journal of the American Statistical Association*, 103(483): 1119–1130. http://dx.doi.org/10.1198/016214508000000689. 586

Guinness, J. (2016). "Permutation Methods for Sharpening Gaussian Process Approximations." https://arxiv.org/abs/1609.05372. 604, 605

Guyon, X. (1995). *Random Fields on a Network: Modeling, Statistics, and Applications*. New York: Springer-Verlag. 585

Higdon, D. (1998). "A process-convolution approach to modeling temperatures in the north Atlantic Ocean." *Environmental and Ecological Statistics*, 5: 173–190. 587

Higdon, D. (2002a). "Space and Space Time Modeling using Process Convolutions." In Anderson, C., Barnett, V., Chatwin, P., and El-Shaarawi, A. (eds.), *Quantitative Methods for Current Environmental Issues*, 37–56. Springer. 586

Higdon, D. (2002b). "Space and Space Time Modeling using Process Convolutions." In Anderson, C., Barnett, V., Chatwin, P., and El-Shaarawi, A. (eds.), *Quantitative Methods for Current Environmental Issues*, 37–56. Springer. MR2059819. 587

Higdon, D., Swall, J., and Kern, J. (1999). "Non-stationary spatial modeling." In Bernardo, J., Berger, J., Dawid, A., and Smith, A. (eds.), *Bayesian Statistics 6*, 761–768. Oxford: Oxford University Press. 587

Hodges, J. S. (2013). *Richly Parameterized Linear Models: Additive, Time Series, and Spatial Models Using Random Effects*. Chapman & Hall/CRC Texts in Statistical Science. Boca Raton, FL: Chapman & Hall/CRC. 588

Hoffman, M. D. and Gelman, A. (2014). "The No U-Turn Sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo." *Journal of Machine Learning Research*, 15: 1593–1623. MR3214779. 605

Kammann, E. E. and Wand, M. P. (2003). "Geoadditive models." *Applied Statistics*, 52: 1–18. MR1963210. doi: http://dx.doi.org/10.1111/1467-9876.00385. 586

Katzfuss, M. (2013). "Bayesian nonstationary modeling for very large spatial datasets." *Environmetrics*, 24: 189–200. 587, 592

Katzfuss, M. (2017). "A multi-resolution approximation for massive spatial datasets." *Journal of the American Statistical Association*, 112: 201–214. http://dx.doi.org/10.1080/01621459.2015.1123632. 593, 606

Katzfuss, M. and Cressie, N. (2012). "Bayesian hierarchical spatio-temporal smoothing for very large datasets." *Environmetrics*, 23: 94–107. MR2873787. doi: http://dx.doi.org/10.1002/env.1147. 587

Kaufman, C. G., Scheverish, M. J., and Nychka, D. W. (2008). "Covariance tapering for likelihood-based estimation in large spatial data sets." *Journal of the American Statistical Association*, 103: 1545–1555. 586, 596

Lark, R. and Papritz, A. (2003). "Fitting a linear model of coregionalization for soil properties using simulated annealing." *Geoderma*, 115: 245–260. 606

Lauritzen, S. L. (1996). *Graphical Models*. Oxford, United Kingdom: Clarendon Press. 597

Lemos, R. and Sansó, B. (2009). "A spatio-temporal model for mean, anomaly and trend fields of North Atlantic Sea surface temperature (with discussion)." *Journal of the American Statistical Association*, 104: 5–25. 586

Lindgren, F., Rue, H., and Lindstrom, J. (2011). "An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(4): 423–498. MR2853727. doi: http://dx.doi.org/10.1111/j.1467-9868.2011.00777.x. 597

Lindley, D. and Smith, A. (1972). "Bayes estimates for the linear model." *Journal of the Royal Statistical Society, Series B*, 34: 1–41. 588

Lopes, H. F., Salazar, E., and Gamerman, D. (2008). "Spatial dynamic factor analysis." *Bayesian Analysis*, 3(4): 759–792. 606

Lopes, H. F. and West, M. (2004). "Bayesian model assessment in factor analysis." *Statistica Sinica*, 14: 41–67. 606

Matheron, G. (1982). "Pour une Analyse Krigeante des Donnes Regionalises." *Centre de Geostatistique*, N 732. 606

Moller, J. and Waagepetersen, R. P. (2003). *Statistical Inference and Simulation for Spatial Point Processes*. Chapman and Hall, first edition. 583

Murphy, K. (2012). *Machine Learning: A probabilistic perspective*. Cambridge, MA: The MIT Press. 597

Neal, R. (2011). "MCMC using Hamiltonian Dynamics." In Brooks, S., Gelman, A., Jones, G. L., and Meng, X.-L. (eds.), *Handbook of Markov Chain Monte Carlo*, 113–162. Boca Raton, FL: CRC Press. 605

Nychka, D., Bandyopadhyay, S., Hammerling, D., Lindgren, F., and Sain, S. (2015). "A multiresolution Gaussian process model for the analysis of large spatial datasets." *Journal of Computational and Graphical Statistics*, 24(2): 579–599. http://dx.doi.org/10.1080/10618600.2014.914946. 587

Nychka, D., Wikle, C., and Royle, J. A. (2002). "Multiresolution models for nonstationary spatial covariance functions." *Statistical Modelling*, 2(4): 315–331. 587

Omidi, M. and Mohammadzadeh, M. (2015). "A new method to build spatio-temporal covariance functions: Analysis of ozone data." *Statistical Papers*, 1–15. MR3557367. doi: http://dx.doi.org/10.1007/s00362-015-0674-2. 602

Paciorek, C. J. and Schervish, M. J. (2006). "Spatial modelling using a new class of nonstationary covariance functions." *Environmetrics*, 483–506. MR2240939. doi: http://dx.doi.org/10.1002/env.785. 587

Quinoñero, C. and Rasmussen, C. (2005). "A unifying view of sparse approximate Gaussian process regression." *Journal of Machine Learning Research*, 6: 1939–1959. 586, 591

Rasmussen, C. E. and Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning*. Cambridge, MA: The MIT Press, first edition. 588, 591

Ren, Q. and Banerjee, S. (2013). "Hierarchical factor models for large spatially misaligned datasets: A low-rank predictive process approach." *Biometrics*, 69: 19–30. 607

Robert, C. and Casella, G. (2004). *Monte Carlo Statistical Methods*. Boca Raton, FL: CRC Press, second edition. 585, 605, 607

Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory and Applications*. Monographs on statistics and applied probability. Boca Raton, FL: Chapman & Hall/CRC. http://opac.inria.fr/record=b1119989 MR2130347. doi: http://dx.doi.org/10.1201/9780203492024. 586, 597

Rue, H., Martino, S., and Chopin, N. (2009). "Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(2): 319–392. http://dx.doi.org/10.1111/j.1467-9868.2008.00700.x. 597, 605

Ruppert, W. M., D. and Carroll, R. (2003). *Semiparametric Regression*. Cambridge, United Kingdom: Cambridge University Press. 590

Sang, H. and Huang, J. Z. (2012). "A full scale approximation of covariance functions for large spatial data sets." *Journal of the Royal Statistical Society, Series B*, 74: 111–132. MR2885842. doi: http://dx.doi.org/10.1111/j.1467-9868.2011.01007.x. 591, 592

Sang, H., Jun, M., and Huang, J. (2011). "Covariance approximation for large multivariate spatial datasets with an application to multiple climate model errors." *Annals of Applied Statistics*, 4: 2519–2548. 592

Sansó, B., Schmidt, A., and Nobre, A. (2008). "Spatio-temporal models based on discrete convolutions." *Canadian Journal of Statistics*, 36: 239–258. MR2522162. doi: http://dx.doi.org/10.1002/cjs.5550360205. 586, 587

Schabenberger, O. and Gotway, C. A. (2004). *Statistical Methods for Spatial Data Analysis*. Chapman and Hall/CRC, first edition. 583

Schmidt, A. M. and Gelfand, A. E. (2003). "A Bayesian coregionalization approach for multivariate pollutant data." *Journal of Geophysical Research*, 108: D24. 606

Shaby, B. A. and Ruppert, D. (2012). "Tapered covariance: Bayesian estimation and asymptotics." *Journal of Computational and Graphical Statistics*, 21: 433–452. MR2945475. doi: http://dx.doi.org/10.1080/10618600.2012.680819. 586

Shi, T. and Cressie, N. (2007). "Global Statistical analysis of MISR aerosol data: A massive data product from NASA's Terra satellite." *Environmetrics*, 18: 665–680. MR2408937. doi: http://dx.doi.org/10.1002/env.864. 587

Stein, M. L. (1999). *Interpolation of Spatial Data: Some Theory for Kriging*. Springer, first edition. MR1697409. doi: http://dx.doi.org/10.1007/978-1-4612-1494-6. 583, 584, 585

Stein, M. L. (2007). "Spatial variation of total column ozone on a global scale." *Annals of Applied Statistics*, 1: 191–210. 586

Stein, M. L. (2008). "A modeling approach for large spatial datasets." *Journal of the Korean Statistical Society*, 37: 3–10. 586

Stein, M. L. (2013). "On a class of space-time intrinsic random functions." *Bernoulli*, 19(2): 387–408. http://dx.doi.org/10.3150/11-BEJ405. 602

Stein, M. L. (2014). "Limitations on low rank approximations for covariance matrices of spatial data." *Spatial Statistics*, 8: 1–19. MR3326818. doi: http://dx.doi.org/10.1016/j.spasta.2013.06.003. 595

Stein, M. L., Chi, Z., and Welty, L. J. (2004). "Approximating likelihoods for large

spatial data sets." *Journal of the Royal Statistical Society, Series B*, 66: 275–296. 586, 600, 601, 602, 604

Stroud, J. R., Stein, M. L., and Lysen, S. (2017). "Bayesian and Maximum Likelihood Estimation for Gaussian Processes on an Incomplete Lattice." *Journal of Computational and Graphical Statistics*, 26: 108–120. doi: http://dx.doi.org/10.1080/10618600.2016.1152970. 600

Sun, Y., Li, B., and Genton, M. (2011). "Geostatistics for large datasets." In Montero, J., Porcu, E., and Schlather, M. (eds.), *Advances and Challenges in Space-time Modelling of Natural Events*, 55–77. Berlin Heidelberg: Springer-Verlag. 585

Vecchia, A. V. (1988). "Estimation and model identification for continuous spatial processes." *Journal of the Royal Statistical Society, Series B*, 50: 297–312. MR0964183. 586, 600, 601, 604

Vecchia, A. V. (1992). "A new method of prediction for spatial regression models with correlated errors." *Journal of the Royal Statistical Society, Series B*, 54: 813–830. MR1185224. 586

Wang, F. and Wall, M. M. (2003). "Generalized common spatial factor model." *Biostatistics*, 4(4): 569–582. http://dx.doi.org/10.1093/biostatistics/4.4.569. 606

Whittle, P. (1954). "On stationary processes in the plane." *Biometrika*, 41: 434–449. 585

Wikle, C. and Cressie, N. (1999). "A dimension reduced approach to space-time Kalman filtering." *Biometrika*, 86: 815–829. 586

Wikle, C. K. (2010). "Low-Rank Representations for Spatial Processes." *Handbook of Spatial Statistics*, 107–118. Gelfand, A. E., Diggle, P., Fuentes, M. and Guttorp, P., editors, Chapman and Hall/CRC, pp. 107–118. 587

Zhang, H. (2007). "Maximum-likelihood estimation for multivariate spatial linear coregionalization models." *Environmetrics*, 18: 125–139. 606

Acknowledgments

The author wishes to thank the Editor-in-Chief (Professor Bruno Sansó) and the anonymous reviewers for very constructive and insightful feedback. In addition, the author also wishes to thank Dr. Abhirup Datta, Dr. Andrew O. Finley and Ms. Lu Zhang for useful discussions. The work of the author was supported in part by NSF DMS-1513654 and NSF IIS-1562303.