Received: 11 April 2022 | Revised: 29 June 2022 | Accepted: 11 July 2022

DOI: 10.1002/env.2748

RESEARCH ARTICLE

WILEY

Conjugate sparse plus low rank models for efficient Bayesian interpolation of large spatial data

Shinichiro Shirota[1] | **Andrew O. Finley**[2] | **Bruce D. Cook**[3] | **Sudipto Banerjee**[4]

[1]Center for the Promotion of Social Data Science Education and Research, Hitotsubashi University, Tokyo, Japan

[2]Departments of Forestry and Geography, Michigan State University, East Lansing, Michigan, USA

[3]Goddard Space Flight Center, National Aeronautics and Space Administration, Greenbelt, Maryland, USA

[4]Department of Biostatistics, University of California, Los Angeles, California, USA

Correspondence
Sudipto Banerjee, Departments of Biostatistics and Statistics, University of California, 650 Charles E. Young Drive South, Los Angeles, CA, USA.
Email: sudipto@ucla.edu

Funding information
Division of Mathematical Sciences, Grant/Award Numbers: NSF/DMS-1513654, NSF/DMS-1513481; NSF/EF, Grant/Award Numbers: EF-1137309, EF-1241874, EF-1253225; NSF/IIS, Grant/Award Number: NSF/IIS-1562303; National Institute of Environmental Health Sciences, Grant/Award Number: NIH/NIEHS-1R01ES027027

Abstract

A key challenge in spatial data science is the analysis for massive spatially-referenced data sets. Such analyses often proceed from Gaussian process specifications that can produce rich and robust inference, but involve dense covariance matrices that lack computationally exploitable structures. Recent developments in spatial statistics offer a variety of massively scalable approaches. Bayesian inference and hierarchical models, in particular, have gained popularity due to their richness and flexibility in accommodating spatial processes. Our current contribution is to provide computationally efficient exact algorithms for spatial interpolation of massive data sets using scalable spatial processes. We combine low-rank Gaussian processes with efficient sparse approximations. Following recent work by Zhang et al. (2019), we model the low-rank process using a Gaussian predictive process (GPP) and the residual process as a sparsity-inducing nearest-neighbor Gaussian process (NNGP). A key contribution here is to implement these models using exact conjugate Bayesian modeling to avoid expensive iterative algorithms. Through the simulation studies, we evaluate performance of the proposed approach and the robustness of our models, especially for long range prediction. We implement our approaches for remotely sensed light detection and ranging (LiDAR) data collected over the US Forest Service Tanana Inventory Unit (TIU) in a remote portion of Interior Alaska.

KEYWORDS

full scale approximations, Gaussian predictive processes, hierarchical models, nearest-neighbor Gaussian processes, scalable spatial models

1 | INTRODUCTION

Statisticians and data scientists working on spatial data analysis are increasingly confronting massive data sets collected over locations numbering in the millions. Often, the primary goal is to carry out spatial predictions at arbitrary locations while accounting for spatially varying predictors and inherent spatial dependencies. Spatial regression models incorporating spatial processes are rich and flexible, but computationally expensive and struggle to scale up to data sets with locations in the order of 10^4—let alone 10^6 which is typical in remote sensing data applications. There is already a substantial literature on modeling large and massive spatial data sets. Insightful reviews of this literature from different perspectives can be found in Sun et al. (2012), Banerjee (2017), and Heaton et al. (2019). The “contest” paper by Heaton et al. (2018) compared the efficiency of predictive inference for a variety of scalable approaches with application in statistical *gap-filling* of remotely sensed data. While some discrepancies between existing methods were noted, most were found to be competitive. Hence, rather than focusing on the superiority of inferential performance, our aim here is to exploit closed-form posterior inference for a flexible class of models.

We outline a practical approach for implementing “full scale approximation” models (e.g., Sang & Huang, 2012; Zhang, Sang, et al., 2019). Such models represent the dependent outcome, or response, variable as the sum of a low-rank spatial process and sparse spatial process. We call this the SLGP (Sparse plus Low-rank Gaussian Process). A key feature of our formulation is that we completely avoid iterative algorithms such as Markov chain Monte Carlo (MCMC), the quadrature-based Integrated Nested Laplace Approximation (Rue et al., 2009), or variational Bayesian methods (Ren et al., 2011). Instead, we formulate conjugate Bayesian models that exploit exact distribution theory to carry out inference (including estimation and prediction with uncertainty quantification). This is especially beneficial for massive data sets of the magnitude we consider here. To achieve this, we use judicious exploratory data analysis with a spatial variogram to learn about the spatial range and the noise-to-signal ratio in the data. We then combine inference from the exact posteriors for a set of fixed values of the spatial range and noise-to-signal ratio (learned from the variogram) and then use K-fold cross-validation (Finley et al., 2019) and present the analysis corresponding to the smallest mean squared prediction errors.

The work presented here was motivated by the practical need to provide computationally efficient prediction with full uncertainty quantification of light detection and ranging (LiDAR) variables for Interior Alaska as part of a National Aeronautics and Space Administration (NASA) Carbon Monitoring System (CMS) program. Current and anticipated sample-based LiDAR data collection campaigns, such as ICESat-2 (Abdalati et al., 2010; ICESat-2, 2015), Global Ecosystem Dynamics Investigation LiDAR (GEDI, 2014), and NASA Goddard’s LiDAR, Hyperspectral, and Thermal (G-LiHT) Airborne Imager (Cook et al., 2013), provide only partial data coverage over domains of interest. Our current CMS work using G-LiHT, and that of future mapping/estimation initiatives with this and other sample-based LiDAR systems, require complete-coverage of LiDAR variable inputs and spatially-explicit uncertainty quantification, hence the need for statistically robust and computationally tractable prediction methods. Such methods should accommodate potentially nonstationary spatial processes which we anticipate for large regions such as the Interior Alaska study area.

Following Zhang, Sang, et al. (2019), we express the original spatial process as a sum of two processes: a low-rank Gaussian predictive process (GPP) (Banerjee et al., 2008) and a sparsity-inducing Nearest-Neighbor Gaussian Process (NNGP) (Datta, Banerjee, Finley, & Gelfand, 2016) for the residual process. The low-rank process captures long-range dependence and smoother variations, while the residual process captures variations at finer scales. We note that full scale approximation models have been formulated in alternative ways. Finley et al., (2009), Sang and Huang (2012), Katzfuss (2017) and Zhang, Sang, et al. (2019) all use the GPP for the low-rank component but differ in how they approximate the residual process. Finley et al. (2009) approximate the residual process using an independent process for adjusting biases in the variance, Sang and Huang (2012) use covariance tapering to introduce shorter-range dependence in the residual process, Katzfuss (2017) use the GPP recursively to construct a multiresolution approximation and Zhang, Sang, et al. (2019) use a NNGP (Datta, Banerjee, Finley, & Gelfand, 2016).

The independent process in Finley et al. (2009) is not equipped to capture short-range dependence and, hence, less efficient than the tapered residuals in Sang and Huang (2012), while the covariance tapering for residuals tends to exhibit greater shorter range dependence than the original process. Furthermore, for the very large data sets, the tapered covariance matrices will still have many nonzero elements, even with a moderate or small range for tapering function, so the computational benefits of sparsity are tempered. Another related matter is that determinant computations for covariance-tapered sparse matrices, in general, can be more complicated and less suited for Bayesian inference.

More recently, Ma and Kang (2017) propose a full scale approach with Markov random field approximations for the residual process. Their approach renders substantial computational benefits and is effective for parameter estimation, but less suited for predictive inference since the Markov random field approximations need not yield a well-defined stochastic process over the entire domain. Also, their approach is not based on a marginal covariance decomposition, that is, decomposition into a predictive process and a residual process, but instead includes two independent processes: one for global dependence and the other for local dependence. While this specification is flexible, two independent processes might result in over-fitting, and less accurate prediction. The NNGP can help resolve some of these issues.

The NNGP’s role as an efficient Bayesian model relies on the well-established accuracy and computational scalability of an approximation attributed to Vecchia in Vecchia (1988), which has also been demonstrated by several authors including, more recently, by Guinness (2018). However, a criticism for Vecchia's approximation is its dependence on the order of locations (Guinness, 2018). Furthermore, this influence is exacerbated especially when the random field is too smooth because distant locations can have non-negligible impact on the value of the process at a given location under strong spatial dependence.

The format of the article is as follows. Section 2 outlines our full scale approximation model in the context of spatial regression models, Section 3 presents some details on the computations pertaining to Bayesian inference, Section 4 offers simulation studies to demonstrate the recovery of parameters and spatial surface and investigate predictive performance, and Section 5 applies our model to the forest canopy height data in Alaska. Finally, Section 6 offers some discussion and future work.

2 | SPARSE PLUS LOW-RANK GAUSSIAN PROCESS MODEL

Consider a set of observed locations $\mathcal{S} = \{\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_n\}$ and a response variable $y(\mathbf{s}_i)$ at location $\mathbf{s}_i \in \mathcal{D} \subseteq \mathbb{R}^d$ in a spatial regression model,

$$y(\mathbf{s}_i) = \mathbf{x}(\mathbf{s}_i)^\top \boldsymbol{\beta} + w(\mathbf{s}_i) + \epsilon(\mathbf{s}_i), \quad \text{for} \quad i = 1, 2, \dots, n, \tag{1}$$

where $\mathbf{x}(\mathbf{s}_i)$ is a fixed $p \times 1$ vector of known spatially-referenced predictors, $w(\mathbf{s}_i)$ is a zero mean spatial process, that is, $\mathbf{w}(\mathcal{S}) \sim \mathcal{N}(\mathbf{0}, \mathbf{C}(\mathcal{S}; \boldsymbol{\theta}))$ where $\mathbf{C}(\mathcal{S}; \boldsymbol{\theta})$ is a $n \times n$ spatial covariance matrix and $\epsilon(\mathbf{s}_i) \sim \mathcal{N}(0, \tau^2)$ is a white noise process capturing measurement error or micro-scale variability with τ^2.

A simple computationally efficient low rank approximation is the GPP, which projects the original process $w(\mathbf{s})$ onto a subspace defined by the realizations of $w(\mathbf{s})$ on the set of knot locations $\mathcal{S}^* = \{\mathbf{s}_1^*, \mathbf{s}_2^*, \dots, \mathbf{s}_r^*\}$. The number of locations (r) is smaller than (n), the computational cost for evaluating likelihood is $\mathcal{O}(nr^2)$ and requires $\mathcal{O}(nr)$ dynamic memory. The low rank process is smoother than the original process, so the variance of the residual in (1) tends to be overestimated. Full scale approximations attempt to mitigate the effects of oversmoothing by decomposing $w(\mathbf{s})$ as a sum of a low-rank process and a finer scale "residual" process, that is, $w(\mathbf{s}) = w_{GPP}(\mathbf{s}) + w_{res}(\mathbf{s})$. This is analogous to traditional regression—we are regressing the spatial process toward (or projecting it onto) a $w_{GPP}(\mathbf{s})$ while the "residual" process $w_{res}(\mathbf{s})$ represents what remains after the regression. We develop this further by modeling $w_{res}(\mathbf{s})$ using an NNGP, refer to the resulting process as the SLGP, and introduce fast algorithms for estimating conjugate SLGP models. The NNGP is a process-based extension to the Vecchia's approximation of a Gaussian process likelihood (Vecchia, 1988) and a number of alternate choices subsequently developed (Katzfuss & Guinness, 2021) can also be used to model $w_{res}(\mathbf{s})$.

2.1 | Latent SLGP models

Like Sang et al. (2011) and Sang and Huang (2012), we use the GPP (Banerjee et al., 2008) as the low-rank component, but unlike covariance tapering or MRFs (Ma & Kang, 2017) we follow Zhang, Sang, et al. (2019) and model the residual process using an NNGP (Datta, Banerjee, Finley, & Gelfand, 2016). We first write $w(\mathbf{s}) \approx w_{GPP}(\mathbf{s}) + w_{res,NNGP}(\mathbf{s})$. We look closer into each of these components.

The GPP is derived as the conditional expectation $w_{GPP}(\mathbf{s}) = \mathrm{E}[w(\mathbf{s})|\mathbf{w}^*]$, where $\mathcal{S}^* = \{\mathbf{s}_1^*, \mathbf{s}_2^*, \dots, \mathbf{s}_r^*\}$ are a set of r locations or "knots" with r considerably smaller than n, and $\mathbf{w}^*$ is an $r \times 1$ vector with elements $w(\mathbf{s}_i^*)$. Thus, $w_{GPP}(\mathbf{s})$ is the orthogonal projection of the parent process $w(\mathbf{s})$ on to its realizations over the set of knots. Using linearity of Gaussian processes, we write $w_{GPP}(\mathbf{s}) = \mathbf{H}(\mathbf{s}; \boldsymbol{\theta})\mathbf{w}^*$, where $\mathbf{H}(\mathbf{s}; \boldsymbol{\theta}) = \mathbf{C}(\mathbf{s}, \mathcal{S}^*; \boldsymbol{\theta})\mathbf{C}(\mathcal{S}^*; \boldsymbol{\theta})^{-1}$, $\mathbf{C}(\mathbf{s}, \mathcal{S}^*; \boldsymbol{\theta})$ is a $1 \times r$ covariance vector and $\mathbf{C}(\mathcal{S}^*; \boldsymbol{\theta})$ is the $r \times r$ covariance matrix. A particular advantage of the GPP is that the residual process $w_{res}(\mathbf{s}) = w(\mathbf{s}) - w_{GPP}(\mathbf{s})$ is tractable. The residual covariance function is

$$\Gamma(\mathbf{s}, \mathbf{s}'; \boldsymbol{\theta}) = C(\mathbf{s}, \mathbf{s}'; \boldsymbol{\theta}) - \mathbf{C}(\mathbf{s}, \mathcal{S}^*; \boldsymbol{\theta})\mathbf{C}(\mathcal{S}^*; \boldsymbol{\theta})^{-1}\mathbf{C}(\mathcal{S}^*, \mathbf{s}; \boldsymbol{\theta}) \; .$$

Our SLGP specification will be completed by approximating this exact residual process by a computationally efficient NNGP. Thus, we first project the process realizations onto realizations over a smaller set of knots, and then approximate the residual using a sparsity-inducing process.

The NNGP is a sparsity inducing GP built from Vecchia-type approximations over a reference set of locations. Let $\mathcal{R} = \{\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_n\}$ be an *ordered* set of locations in our domain and define neighbor sets $N(\mathbf{r}_i)$ to be the set of its m nearest neighbors from the locations that precede it, that is, from $\{\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_{i-1}\}$. In fact, unlike the set of "knots" for GPPs, the reference set can be as large as needed to construct a suitable approximation for the process. The key idea here is to approximate the $n \times n$ precision matrix as $\boldsymbol{\Gamma}(\mathcal{R};\boldsymbol{\theta})^{-1} \approx (\mathbf{I}-\mathbf{A})^{\top}\mathbf{F}^{-1}(\mathbf{I}-\mathbf{A})$, where $\mathbf{A}$ and $\mathbf{F}$ depend on $\boldsymbol{\theta}$, but we suppress this dependence for simplicity in the subsequent notation. Here, $\mathbf{F}^{-1}$ is an $n \times n$ diagonal matrix with $[\mathbf{F}^{-1}]_{ii} = \mathrm{var}\{w(\mathbf{r}_i)|w(N(\mathbf{r}_i))\} = \Gamma(\mathbf{r}_i, \mathbf{r}_i;\boldsymbol{\theta}) - \boldsymbol{\Gamma}(\mathbf{r}_i, N(\mathbf{r}_i);\boldsymbol{\theta})\boldsymbol{\Gamma}(N(\mathbf{r}_i);\boldsymbol{\theta})^{-1}\boldsymbol{\Gamma}(N(\mathbf{r}_i), \mathbf{r}_i;\boldsymbol{\theta})$ and $\mathbf{A}$ is an $n \times n$ sparse lower-triangular such that $[\mathbf{A}]_{ij} = 0$ for all $j \geq i$ and $\{[\mathbf{A}]_{ij} : j = 1, 2, \dots, i-1\}$ are kriging weights of $w(\mathbf{r}_i)$ based upon $w(N(\mathbf{r}_i))$, that is, for $i = 1, 2, \dots, n$ and $j = 1, 2, \dots, i-1$

$$[\mathbf{A}]_{ij} = \boldsymbol{\Gamma}(\mathbf{r}_i, N(\mathbf{r}_i);\boldsymbol{\theta})\boldsymbol{\Gamma}(N(\mathbf{r}_i);\boldsymbol{\theta})^{-1}. \tag{2}$$

With this set up, the NNGP is defined recursively as $w_{res,NNGP}(\mathbf{r}_1) = w(\mathbf{r}_1)$ and $w_{res,NNGP}(\mathbf{r}_i) = \sum_{j=1}^{i-1}\mathbf{A}_{\theta,ij}w_{res,NNGP}(\mathbf{r}_j) + \eta(\mathbf{r}_i)$ for $i = 2, 3, \dots, n$, where each $\eta(\mathbf{r}_i) \overset{ind}{\sim} N(0, [\mathbf{F}^{-1}]_{ii})$. This specifies the realizations over the reference set. The definition is extended to arbitrary locations using $\mathbf{w}_{res,NNGP}(\mathbf{s}) = \sum_{j=1}^{N}\mathbf{A}_{\theta,j}(\mathbf{s})w(\mathbf{r}_i) + \boldsymbol{\eta}(\mathbf{s})$ for any location $\mathbf{s}$ outside of $\mathcal{R}$, where $\mathbf{A}_{\theta,j}(\mathbf{s}) = \boldsymbol{\Gamma}(\mathbf{s}, N(\mathbf{s});\boldsymbol{\theta})\boldsymbol{\Gamma}(N(\mathbf{s});\boldsymbol{\theta})^{-1}$ where $\mathrm{Pa}(\mathbf{s})$ is the set of m nearest neighbors of $\mathbf{s}$ from within the locations in $\mathcal{R}$. Sparsity is induced because the number of nonzero elements in $\mathbf{A}$ is limited to no more than m in each row.

While $w_{GPP}(\mathbf{s})$ is a low-rank process that yields degenerate finite-dimensional probability laws on realizations over sets where the number of spatial locations exceed the number of knots, the $w_{res,NNGP}(\mathbf{s})$ is a sparse full-rank process that, by construction, will always yield nondegenerate finite-dimensional probability laws. Therefore, the process $w(\mathbf{s}) = w_{GPP}(\mathbf{s}) + w_{res,NNGP}(\mathbf{s})$ is also nondegenerate. We will call modeling the latent process using SLGP as the latent SLGP model. This has at least one important modeling implication: we can use $w(\mathbf{s})$ to model the outcomes themselves. In particular, we can devise a *response* SLGP model, where rather than modeling latent spatial process, or regression coefficients, we apply SLGP to model the response itself. The response model is especially convenient for constructing conjugate Bayesian SLGPs for which we can devise fast algorithms for massive data sets.

2.2 | Conjugate SLGP models

The marginal, or collapsed, likelihood obtained by integrating out the $\mathbf{w}$ from (1) yields

$$y(\mathbf{s}_i) = \mathbf{x}(\mathbf{s}_i)^{\top}\boldsymbol{\beta} + \epsilon(\mathbf{s}_i), \quad \text{for} \quad i = 1, 2, \dots, n \tag{3}$$

and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}(\mathcal{S};\boldsymbol{\theta}) + \tau^2\mathbf{I})$, where $\boldsymbol{\epsilon}$ is the $n \times 1$ vector with entries $\epsilon(\mathbf{s}_i)$. We set $\alpha = \tau^2/\sigma^2$ and rewrite the marginal model as $N(\mathbf{y}|\mathbf{X}\boldsymbol{\beta}, \sigma^2\mathbf{M}(\mathcal{S};\boldsymbol{\theta}))$, where $\mathbf{M}(\mathcal{S};\boldsymbol{\theta}) = \mathbf{R}(\mathcal{S};\boldsymbol{\theta}) + \alpha\mathbf{I}$ and $\mathbf{R}(\mathcal{S};\boldsymbol{\theta}) = \mathbf{C}(\mathcal{S};\boldsymbol{\theta})$ denotes the spatial correlation matrix. We suppress the dependence of these matrices on α in the subsequent development.

We decompose $\mathbf{M}(\mathcal{S};\boldsymbol{\theta})$ into low rank and sparse matrices,

$$\mathbf{M}(\mathcal{S};\boldsymbol{\theta}) = \mathbf{J}(\mathcal{S}, \mathcal{S}^*;\boldsymbol{\theta})\mathbf{R}(\mathcal{S}^*)\mathbf{J}(\mathcal{S}, \mathcal{S}^*;\boldsymbol{\theta})^{\top} + \boldsymbol{\Omega}(\mathcal{S};\boldsymbol{\theta}), \tag{4}$$

where $\mathbf{J}(\mathcal{S}, \mathcal{S}^*;\boldsymbol{\theta}) = \mathbf{R}(\mathcal{S}, \mathcal{S}^*;\boldsymbol{\theta})\mathbf{R}(\mathcal{S}^*;\boldsymbol{\theta})^{-1}$ and $\boldsymbol{\Omega}(\mathcal{S};\boldsymbol{\theta}) = \mathbf{M}(\mathcal{S};\boldsymbol{\theta}) - \mathbf{J}(\mathcal{S}, \mathcal{S}^*;\boldsymbol{\theta})\mathbf{R}(\mathcal{S}^*;\boldsymbol{\theta})\mathbf{J}(\mathcal{S}, \mathcal{S}^*;\boldsymbol{\theta})^{\top}$ is a residual covariance matrix. Since $\mathbf{M}(\mathcal{S};\boldsymbol{\theta})$ and $\mathbf{R}(\mathcal{S}^*;\boldsymbol{\theta})$ are necessarily a positive definite (having been constructed from the spatial covariance matrix) and $\mathbf{J}(\mathcal{S}, \mathcal{S}^*;\boldsymbol{\theta})\mathbf{R}(\mathcal{S}^*;\boldsymbol{\theta})\mathbf{J}(\mathcal{S}, \mathcal{S}^*;\boldsymbol{\theta})^{\top}$ is non-negative definite, $\boldsymbol{\Omega}(\mathcal{S};\boldsymbol{\theta})$ is also positive definite. For the conjugate SLGP, we rewrite the above model as $N(\mathbf{y}|\mathbf{X}\boldsymbol{\beta} + \mathbf{J}(\mathcal{S}, \mathcal{S}^*;\boldsymbol{\theta})\mathbf{z}^*, \sigma^2\boldsymbol{\Omega}(\mathcal{S};\boldsymbol{\theta}))$, where $\mathbf{z}^* \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{R}(\mathcal{S}^*;\boldsymbol{\theta}))$ is defined on $\mathcal{S}^*$. We need to sample $\mathbf{z}^*$ on $\mathcal{S}^*$ in addition to $\boldsymbol{\beta}$ and σ^2. $\mathbf{z}^*$ can be sampled by extending the definition of $\boldsymbol{\beta}$. We extend the definition of $\boldsymbol{\beta}$ to $\boldsymbol{\beta}^*$ as

$$\boldsymbol{\beta}^* = \begin{pmatrix} \boldsymbol{\beta} \\ \mathbf{z}^* \end{pmatrix}, \quad \boldsymbol{\beta}^* \sim \mathcal{N}(\boldsymbol{\mu}_{\beta^*}, \sigma^2\mathbf{V}_{\beta^*}), \quad \boldsymbol{\mu}_{\beta^*} = \begin{pmatrix} \boldsymbol{\mu}_{\beta} \\ \boldsymbol{\mu}_{z^*} \end{pmatrix} \quad \text{and} \quad \mathbf{V}_{\beta^*} = \begin{pmatrix} \mathbf{V}_{\beta} & \mathbf{O} \\ \mathbf{O} & \mathbf{R}(\mathcal{S}^*) \end{pmatrix}. \tag{5}$$

The nearest-neighbor approximation is implemented for $\boldsymbol{\Omega}(\mathcal{S};\boldsymbol{\theta})$ instead of $\mathbf{M}(\mathcal{S};\boldsymbol{\theta})$. For fixed $\{\alpha, \phi\}$, the likelihood and prior are $\mathcal{IG}(\sigma^2|a_{\sigma}, b_{\sigma})\mathcal{N}(\boldsymbol{\beta}^*|\boldsymbol{\mu}_{\beta^*}, \sigma^2\mathbf{V}_{\beta^*})\mathcal{N}(\mathbf{y}|\mathbf{X}^*\boldsymbol{\beta}^*, \sigma^2\tilde{\boldsymbol{\Omega}})$ where $\mathbf{X}^* = (\mathbf{X}, \mathbf{J}(\mathcal{S}, \mathcal{S}^*;\boldsymbol{\theta}))$. The sampling algorithm and its sampling costs closely follow that of the conjugate NNGP in Finley et al. (2019). The dimension of $\boldsymbol{\beta}^*$ is $p + r$ where p is the number of parameters and r is the number of knots; $\boldsymbol{\beta}^*$ includes the low dimensional GP $\mathbf{z}^*$ on knots $\mathcal{S}^*$ with covariance matrix $\sigma^2\mathbf{R}(\mathcal{S}^*)$. In the likelihood, the covariance matrix now is $\sigma^2\tilde{\boldsymbol{\Omega}}$ from $\sigma^2\tilde{\mathbf{M}}$ where $\tilde{\boldsymbol{\Omega}}$ is the nearest-neighbor approximation of $\boldsymbol{\Omega}$. Hence,

$$p(\boldsymbol{\beta}^*, \sigma^2|\mathbf{y}) \propto \mathcal{IG}(\sigma^2|a_\sigma^*, b_\sigma^*) \times \mathcal{N}(\boldsymbol{\beta}^*|\mathbf{B}^{-1}\mathbf{b}, \sigma^2\mathbf{B}^{-1}), \tag{6}$$

where $a_\sigma^* = a_\sigma + n/2$, $b_\sigma^* = b_\sigma + \frac{1}{2}(\boldsymbol{\mu}_{\beta^*}^\top \mathbf{V}_{\beta^*}^{-1}\boldsymbol{\mu}_{\beta^*} + \mathbf{y}^\top\tilde{\boldsymbol{\Omega}}^{-1}\mathbf{y} - \mathbf{b}^\top\mathbf{B}^{-1}\mathbf{b})$, $\mathbf{B} = \mathbf{V}_{\beta^*}^{-1} + \mathbf{X}^{*\top}\tilde{\boldsymbol{\Omega}}^{-1}\mathbf{X}^*$ and $\mathbf{b} = \mathbf{V}_{\beta^*}^{-1}\boldsymbol{\mu}_{\beta^*} + \mathbf{X}^{*\top}\tilde{\boldsymbol{\Omega}}^{-1}\mathbf{y}$. $\boldsymbol{\mu}_{\beta^*}, \mathbf{V}_{\beta^*}, \mathbf{X}^*, \boldsymbol{\beta}^*, \tilde{\boldsymbol{\Omega}}$ are defined above. The sampling cost for sampling from the conjugate SLGP model is obtained by replacing p in Finley et al. (2019) with $p + r$ and also accounting for the costs in calculating $\mathbf{J}$ and $\boldsymbol{\Omega}$.

3 | IMPLEMENTATION DETAILS FOR SLGP

For sampling $w_{res,SLGP}$, we need to evaluate $\mathcal{N}(\mathbf{w}_{res,SLGP}|\mathbf{0}, \boldsymbol{\Gamma}(\mathcal{S};\boldsymbol{\theta}))$. From the structure of $\mathbf{A}$ it is evident that $\mathbf{I} - \mathbf{A}$ is nonsingular so $\boldsymbol{\Gamma} = (\mathbf{I} - \mathbf{A})^{-1}\mathbf{F}(\mathbf{I} - \mathbf{A})^{-\top}$. Let `a[i,j]`, `f[i,j]` and `Ga[i,j]` denote the (i,j)th entries of $\mathbf{A}$, $\mathbf{F}$ and $\boldsymbol{\Gamma}$, respectively. Note that `f[1,1] = Ga[1,1]` and the first row of $\mathbf{A}$ is $\mathbf{0}^\top$. A pseudo-code to compute the remaining elements of $\mathbf{A}$ and $\mathbf{F}$ is:

```
for(i in 1:n-1){
  a[i+1,1:i] = solve(Ga[1:i,1:i], Ga[1:i,i+1])
  f[i+1,i+1] = Ga[i+1,i+1]
                 - dot(Ga[i+1,1:i],a[i+1,1:i])
}.
```
(7)

Here, `a[i+1,1:i]` is the `1×i` row vector comprising the possibly nonzero elements of the `i+1`th row of $\mathbf{A}$, `Ga[1:i,1:i]` is the `i × i` leading principal submatrix of $\boldsymbol{\Gamma}$, `Ga[1:i, i]` is the `i × 1` row vector formed by the first `i` elements in the `i`th column of $\boldsymbol{\Gamma}$, `Ga[i, 1:i]` is the `1 × i` row vector formed by the first `i` elements in the `i`th row of $\boldsymbol{\Gamma}$, `solve(B,b)` solves the linear system `Bx = b`, and `dot(u,v)` provides the inner product between vectors `u` and `v`. Also, $\det(\boldsymbol{\Gamma}) = \prod_{\mathtt{i=1}}^{\mathtt{n}}$ `f[i,i]` is obtained with almost no additional cost.

The above pseudocode provides a way to obtain the Cholesky decomposition of $\boldsymbol{\Gamma}$. If $\boldsymbol{\Gamma} = \mathbf{LDL}^\top$ is the Cholesky decomposition, then $\mathbf{L} = (\mathbf{I} - \mathbf{A})^{-1}$. 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 $\mathbf{A}$ to be zero. For example, suppose we set at most m elements in each row of $\mathbf{A}$ 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 $\mathbf{A}$ and the diagonal elements of $\mathbf{F}$ much more efficiently as:

```
for(i in 1:n-1){
    Pa = N[i+1] # neighbors of i+1
    a[i+1,Pa] = solve(Ga[Pa,Pa], Ga[(i+1),Pa])
    f[i+1,i+1] = Ga[i+1,i+1]
                   - dot(Ga[(i+1),Pa], a[i+1,Pa])
}.
```
(8)

In (8) we solve $n - 1$ linear systems of size at most $m \times m$. This can be performed in $\mathcal{O}(nm^3)$ flops, whereas the earlier pseudocode in (7) for the dense model required $\mathcal{O}(n^3)$ flops. These computations can be performed in parallel as each iteration of the loop is independent of the others. The density $\mathcal{N}(\mathbf{w}_{res,NNGP}|\mathbf{0}, \tilde{\boldsymbol{\Gamma}})$ is cheap to compute since $\tilde{\boldsymbol{\Gamma}}^{-1}$ is sparse and $\det(\tilde{\boldsymbol{\Gamma}}^{-1})$ is the product of the diagonal elements of $\mathbf{F}^{-1}$.

The factorization of $\tilde{\boldsymbol{\Gamma}}^{-1}$ facilitates cheap computation of quadratic forms $\mathbf{u}^\top\tilde{\boldsymbol{\Gamma}}^{-1}\mathbf{v}$ in terms $\mathbf{A}$ and $\mathbf{F}$. The algorithm to evaluate quadratic forms `qf(u,v,A,F)` is provided in the following pseudocode:

```
qf(u,v,A,F) = u[1] * v[1] / F[1,1]
for(i in 2:n){
   qf(u,v,A,F) = qf(u,v,A,F)
                + (u[i] - dot(A[i,N(i)], u[N(i)]))          (9)
                *(v[i]-dot(A[i,N(i)],v[N(i)]))/F[i,i]
}.
```

Observe (9) only involves inner products of $m \times 1$ vectors. So, the entire for loop can be computed using $\mathcal{O}(nm)$ flops as compared to $\mathcal{O}(n^2)$ flops typically required to evaluate quadratic forms involving an $n \times n$ dense matrix. Also, importantly, the determinant of $\tilde{\boldsymbol{\Gamma}}$ is obtained with almost no additional cost: it is simply $\prod_{\mathtt{i=1}}^{\mathtt{n}} \mathtt{f[i,i]}$.

Hence, while $\tilde{\boldsymbol{\Gamma}}$ need not be sparse, the density $N(\mathbf{w}_{res,NNGP}|\mathbf{0}, \tilde{\boldsymbol{\Gamma}})$ is cheap to compute requiring only $\mathcal{O}(n)$ flops. The Markov chain Monte Carlo (MCMC) implementation of the NNGP model in Datta et al. (Datta, Banerjee, Finley, & Gelfand, 2016) requires updating the n latent spatial effects $\mathbf{w}$ sequentially, in addition to the regression and covariance parameters. While this ensures substantial computational scalability in terms of evaluating the likelihood, the behavior of MCMC convergence for such a high-dimensional model is difficult to study and may well prove unreliable. Finley et al. (2019) reported that, for very large spatial datasets, sequential updating of the random effects often leads to very poor mixing in the MCMC. The computational gains per MCMC iteration is thus offset by a slow converging MCMC. Liu et al. (1994) showed that MCMC algorithms where one or more variables are marginalized out tend to have lower autocorrelation and improved convergence behavior. Here we explore SLGP models that drastically reduce the parameter dimensionality of the SLGP models by marginalizing over the entire vector of spatial random effects. These models are SLGP alternatives for NNGP suggested by Finley et al. (2019) Two different variants are developed and their relative merits and demerits are assessed both in terms of computational burden as well as model prediction and inference.

Implementation and computing

All subsequent analyses were conducted on a Linux workstation with two 18-core Xeon(R) CPU E5-2699 v3 @ 2.30GHz processors and 512 GB of memory. The NNGP and SLGP parameter estimation and prediction algorithms were programmed in `C++` and used `openBLAS` (Zhang, 2016) and Linear Algebra Package (LAPACK; http://www.netlib.org/lapack) for efficient matrix computations. `openBLAS` is an implementation of Basic Linear Algebra Subprograms (BLAS; http://www.netlib.org/blas) capable of exploiting multiple processors. Additional multiprocessor parallelization used `openMP` (Dagum & Menon, 1998) to improve performance of key steps within the algorithms. In particular, substantial gains were realized by distributing the calculation of `A` and `F`, and subsequent calls to the `qf` function over the total number of available cores using the `openMP omp for` directive. Cross-validation and prediction algorithms were also parallelized in a straightforward manner, i.e., each cross-validation fold and prediction for a given location are independent and hence can be spread across cores.

4 | SIMULATION STUDY

We assess the NNGP and SLGP model ability to estimate α and ϕ using a K-fold cross-validation and subsequent out-of-sample prediction performance following the algorithm defined in the Appendix. Simulated data comprising 35,000 outcomes were generated from the Gaussian process plus nugget model within a unit square domain. The generating model was $\mathbf{y} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_S(\boldsymbol{\theta}) + \tau^2\mathbf{I})$ with an exponential covariance function $\boldsymbol{\theta} = (\sigma^2 = 1, \phi = 12)$, and $\tau^2 = 0.5$, that is, $\alpha = 0.5$. These data were divided at random into a training ($n = 25{,}000$) and holdout ($n_0 = 10{,}000$) set. Given the training set, a 5-fold cross-validation was used to find optimal values of ϕ and α under two different scoring rules; root mean squared prediction error (RMSPE) and continuous ranked probability score (CRPS; Gneiting & Raftery, 2007). While CRPS is less common for assessing predictive performance in the remote sensing literature, it has some clear advantages over the more common scoring rules such as RMSPE. As detailed in Gneiting and Raftery (2007), CRPS is attractive for both practical and theoretical reasons, principal among them is that the rule favors models that yield both accurate and precise prediction, whereas RMSPE only considers accuracy. Lower values of RMSPE and CRPS indicate models with

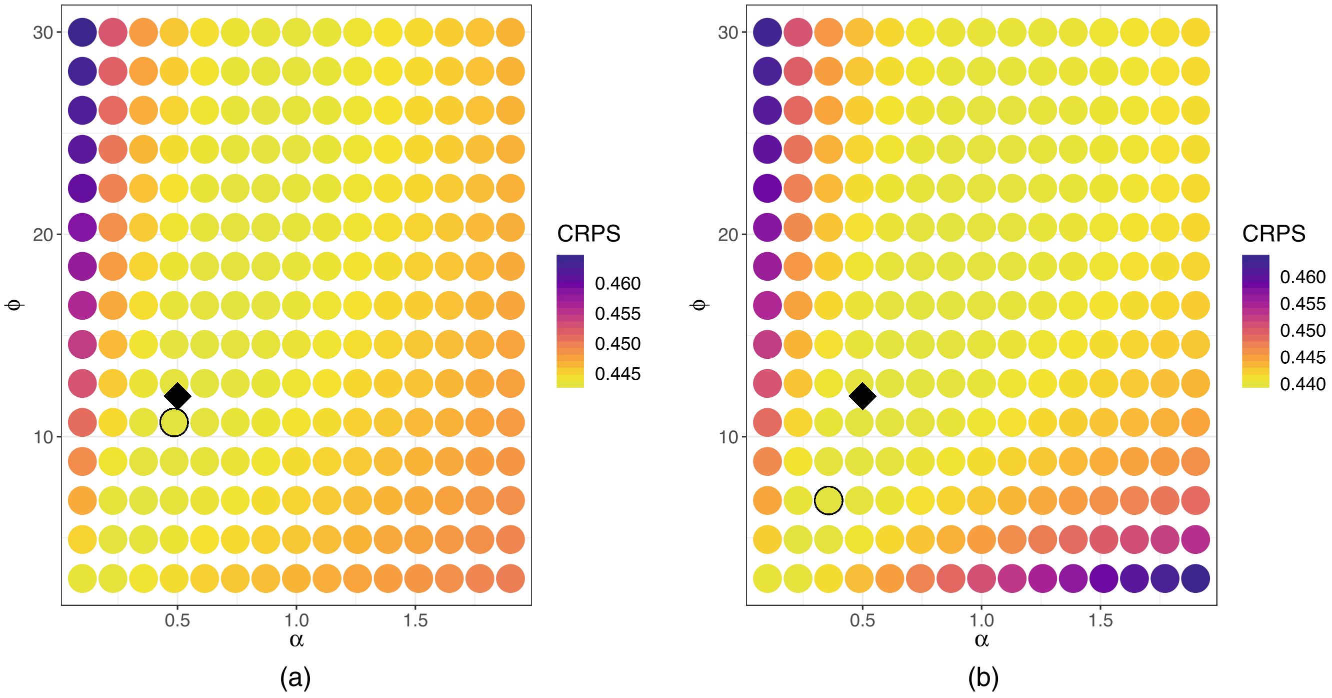


FIGURE 1 NNGP (a) and SLGP (b) CRPS search grid results for the simulated data. The "optimal" parameter combination, that is, yielding the lowest CRPS, is circled. A diamond symbol identifies the "true" α and ϕ used to generate the data.

TABLE 1 Simulated data holdout set cross-validation for RMSPE- and CRPS-based optimized parameter estimates

	RMSPE Optim.		CRPS Optim.	
	NNGP	**SLGP**	**NNGP**	**SLGP**
RMSPE	0.776	0.775	0.776	0.775
CRPS	0.438	0.438	0.438	0.437

better predictive performance. The search grid comprised 225 parameter sets with combinations of 15 α values from 0.1 to 1.9 and 15 ϕ values from 3 and 30. Figure 1 shows the average CRPS from the 5-fold cross-validation for each of the candidate α and ϕ pairs for NNGP and SLGP using a grid of r=100 knots. In Figure 1 the optimal (i.e., minimum) CRPS is circled and the "true" parameter value set used to generate the data is denoted with a diamond. While not shown, the RMSPE-based optimization figures looked nearly identical to Figure 1. Figure 1 also reveals that predictive performance is fairly insensitive to choice of α and ϕ except for under fairly extreme misspecification, e.g., $\alpha \leq 0.1$ and $\alpha > 1.9$ for the NNGP and SLGP and $\phi \leq 3$ for SLGP (regions with purple circles). Using the 5-fold cross-validation identified optimal parameters, the NNGP and SLGP models were used to predict at the 10,000 holdout locations. Results for this out-of-sample prediction are given in Table 1, which shows negligible predictive performance between the two models regardless of the scoring rule used to select the ϕ and α.

5 | FOREST CANOPY HEIGHT MODELS

Digital maps of forest structure are key inputs to many ecosystem and Earth system modeling efforts (Finney, 2004; Hurtt et al., 2004; Klein et al., 2015; Lefsky, 2010; Stratton, 2006). These and similar applications seek inference about forest canopy height variables and predictions that can be propagated through computer models of ecosystem function to yield more robust error quantification. Given the scientific and applied interest in forest structure, there is increasing demand for wall-to-wall forest canopy height data at national and biome scales. To date, information about canopy height has been developed from sparse samples of field measurements and complete-coverage but limited spatial extent LiDAR data (Baccini et al., 2004; Lefsky, 2010; Simard et al., 2011). Next generation LiDAR systems capable of large-scale mapping of forest canopy characteristics, such as ICESat-2 (Abdalati et al., 2010; ICESat-2, 2015), Global Ecosystem Dynamics

Investigation LiDAR (GEDI, 2014), and NASA Goddard's LiDAR, Hyperspectral, and Thermal (G-LiHT) Airborne Imager (Cook et al., 2013), sample forest features using LiDAR instruments in long transects or cluster designs (see, e.g., the strips of LiDAR in Figure 2). These next generation systems yield LiDAR data over the desired large spatial extents; however, the sparseness of the LiDAR sampling designs means prediction is required to deliver the desired wall-to-wall data products.

Our goal is to create high spatial resolution forest canopy height predictions, with accompanying uncertainty estimates, for the US Forest Service Tanana Inventory Unit (TIU) that covers a large portion of Interior Alaska using a sparse sample of LiDAR data from G-LiHT. We assess the proposed conjugate regression models in two settings illustrated in Figure 2: (1) the small areal extent and intensively sampled Bonanza Creek Experimental Forest (BCEF); (2) the large areal extent and sparsely sampled TIU, which contains the BCEF.

5.1 | Study sites

The BCEF domain delineated for this study, Figure 2b, is ~21,000 ha and includes a section of the Tanana River floodplain along the southeastern border. Like the broader TIU, the BCEF is a mixture of non-forest and forest vegetation featuring white spruce, black spruce, tamarack, quaking aspen, and balsam poplar trees mixed with willow and alder shrubland species LTER (n.d.). Figure 2b also shows location of the $n = 188{,}717$ G-LiHT LiDAR forest canopy height measurements.

Figure 2a shows the ~140,000 km^2 TIU study area and location of the $n = 17{,}357, 816$ G-LiHT LiDAR measurements of forest canopy height. Recently, Finley et al. (2019) considered a subset of these data to assess alternate formulations of hierarchical NNGP models for improved convergence, faster computing time, and more robust and reproducible Bayesian inference. Their contribution focused on computational and parameterization improvements for NNGP models, but no effort was directed to exploring regression models beyond a basic spatially-varying intercept with an isotropic stationary covariance. The TIU's forest composition and structure are the result of myriad large and small spatial scale biotic (e.g., insect disturbance) and abiotic (e.g., soil, topography, climate, wind, fire) factors that cause spatially complex mortality and regrowth patterns. The result is a forest canopy that can exhibit both short- and long-range spatial autocorrelation and change in variability across the TIU. Such complexity in the outcome's spatial dependence surface encourages the exploration of more flexible covariance functions such as offered by the SLGP model to define the Gaussian process.

5.2 | Remote sensing data

Forest canopy height predictions were informed using sampled LiDAR canopy height from G-LiHT (outcome variable) and complete-coverage percent tree cover and forest fire occurrence (predictor variables). G-LiHT is a portable multi-sensor system that can be mounted to a fixed wing aircraft. G-LiHT's on-board laser altimeter (VQ-480, Riegl Laser Measurement Systems, Horn, Austria) provides an effective measurement rate of up to 150 kilohertz along a 60°, swath perpendicular to the flight direction using a 1550 nanometer laser. At a nominal flying altitude of 335 m, laser pulse

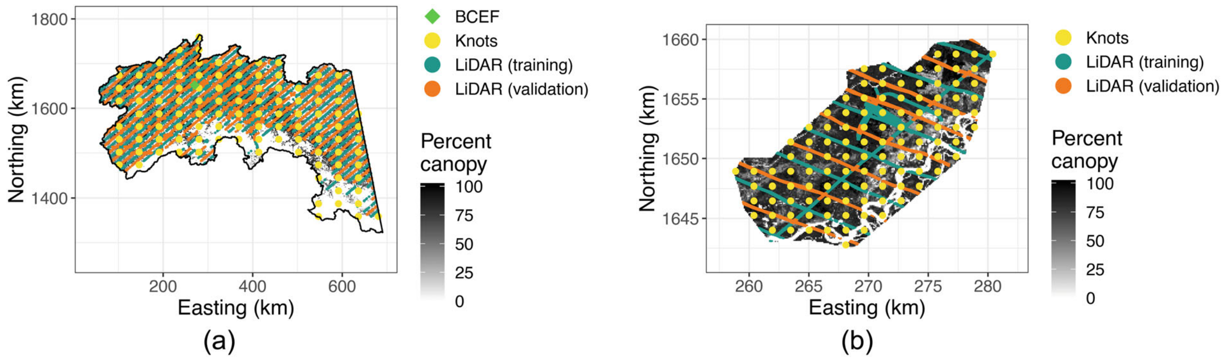


FIGURE 2 (a) Tanana Inventory Unit (TIU) study area. Point symbols denote location of the Bonanza Creek Experimental Forest (BCEF) within the TIU, SLGP knots, and G-LiHT LiDAR measurements of forest canopy height for model training and validation. Underlying grayscale map is the percent tree cover across the TIU. (b) Shows location of knots, G-LiHT LiDAR forest canopy height measurements, and percent tree cover for the BCEF footprints have an approximate 10 cm diameter. The instrument is capable of producing up to eight returns per pulse. Point cloud information was summarized to a 13 × 13 m grid cell size (grid cell area equal to 169 m^2). Over each grid cell, the maximum canopy height was estimated using the 100th percentile height of the point cloud then square root transformed before being used in the subsequent models. G-LiHT data in Summer 2014 for the study areas are available online (G-LiHT: Goddard's LiDAR, 2019). Two predictors that completely cover the TIU were used to help explain variability in forest canopy height. First, a Landsat derived percent tree cover data product developed by Hansen et al. (2013), shown as the gray scale surfaces in Figure 2. This product provides percent tree cover estimates for peak growing season in 2010 (most recent year available) and was created using a regression tree model applied to Landsat 7 ETM+ annual composites. These data are provided by the United States Geological Survey (USGS) on an approximate 30 m grid covering the entire globe (Hansen et al., 2013). Second, the perimeters of past fire events from 1947–2014 were obtained from the Alaska Interagency Coordination Center Alaska fire history data product (AICC, 2016). Forest recovery/regrowth following fire is very slow in Interior Alaska. Hence we discretized the fire history data to 1 if the fire occurred within the past 20 years and 0 otherwise. Fire occurrence had the same value for the entire BCEF and therefore only the percent tree cover predictor was used for the subsequent BCEF> analysis.

5.3 | Analysis

Ultimately we seek models that provide the best prediction at locations that were not sampled by the LiDAR. Using spatial regression models considered here, those locations furthest from observed locations will be most difficult to predict with high accuracy and precision. Given this consideration and the constraints of the LiDAR transect sampling design, we use the following steps to evaluate and arrive upon the "best" models for domain wide prediction:

Step 1: Split LiDAR datasets into a training and holdout set. The holdout set consists of observations from every other LiDAR transect (to ensure a good number of distant predictions) plus a subset of observations within the transects from which the training observations are drawn (to allow for some "nearby" predictions), see Figures 2b and 2a for BCEF and TIU respectively.

Step 2: Use exploratory data analysis (EDA) to identify a range of ϕ and α values over which to conduct a grid search for the "optimal" combination of these parameters.

Step 3: Search the grid defined in Step 2 to identify optimal ϕ and α for NNGP and SLGP models using 5-fold cross-validation of the training set based on CRPS. We reserve the holdout set for subsequent out-of-sampled validation in Step 4.

Step 4: Use NNGP and SLGP specific optimal ϕ and α from Step 3 to predict at holdout set locations, then compare these predictions to holdout set observations using CRPS.

Step 5: Use optimal ϕ and α from Step 4 and all available data (i.e., training plus holdout datasets) to predict for all locations within the domain.

Again, in practice, we might skip splitting the data into training and holdout sets and simply use K-fold cross-validation on all available data to identify the best model and accompanying optimal set of parameters (as illustrated in the simulated data analysis). However, given the unique LiDAR sampling design and the desire to assess how models perform making distant predictions (i.e., between LiDAR tracks), we opted for this more elaborate calibration and model choice scheme.

For Step 2, a semivariogram of the non-spatial regression model residuals can inform how the residual spatial/nonspatial variance (i.e., outcome variance not explained by the regression mean) is partitioned and provide a rough estimate of the spatial range, see, for example, chapter 5 in Banerjee et al. (2014) for details. In the subsequent analyses we use an exponential spatial correlation function that approaches zero as the distance between locations increases. Therefore we define the distance, d_0, at which this correlation drops to 0.05 as the "effective spatial range," which allows us to solve $\phi = -\log(0.05)/d_0$. The semivarogram and empirical parameter estimates (from a nonlinear regression) for the BCEF are given in Figure 4. These estimates suggest a search grid over the intervals $\alpha = (0.01, 1)$ and $\phi = (0.1, 10)$ would be reasonable. Figure 3 depicts this search grid and results from Step 3 that identify the "optimal" set $\phi = 1.53$ and $\alpha = 0.01$ as producing the minimum CRPS for both NNGP and SLGP.

Following Step 4, the model specific "optimal" parameter set was used to predict for holdout set locations, the results of which are given in Table 2 along with the associated fixed and estimated model parameters. Here, in addition to the NNGP and SLGP, we included the parameters and prediction metrics for the non-spatial regression model (NS), which

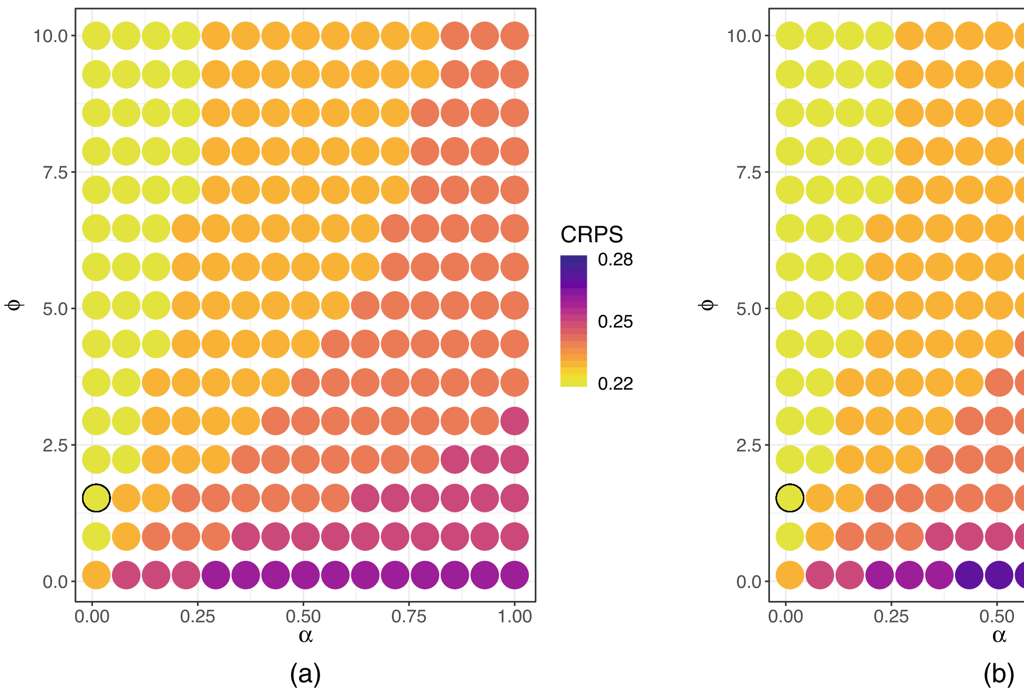


FIGURE 3 NNGP (a) and SLGP (b) CRPS search grid results for the BECF. The "optimal" parameter combination, that is, yielding the lowest CRPS, is circled.

TABLE 2 BCEF holdout set cross-validation CRPS-based parameter estimates and prediction metrics

			SLGP	
	NS	**NNGP**	$r = 110$	$r = 200$
β_0	1.80 (3.2[a])	2.34 (0.045)	2.61 (0.058)	2.71 (0.058)
β_{TC}	0.028 (5.0[b])	0.004 (1.0[c])	0.004 (1.0[c])	0.004 (1.0[c])
α	–	0.01	0.01	0.01
τ^2	0.69	0.04		0.04
σ^2	–	3.76 (1.1[a])	3.76 (1.1[a])	3.76 (1.0[a])
ϕ	–	1.53	1.53	1.53
CRPS	0.49	0.33	0.32	0.32
RMSPE	0.85	0.59	0.59	0.59

Note: Parameter variances estimates given in parentheses. RMSPE values were generated from holdout set cross-validation RMSPE-based parameter estimates that are not reported in this table.

[a] Times 10^{-3}.

[b] Times 10^{-8}.

[c] Times 10^{-5}.

for the BCEF is informed using an intercept and percent tree cover predictor. Also, we considered two different knot grid intensities with which to calculate the SLGP (a depiction of the 200 knot configuration is given in Figure 2b). Here, CRPS and RMSPE values show the NNGP and SLGP models yield improved prediction over the NS model, and there is no appreciable difference between the NNGP and SLGP models. There are a few things to note in the parameter estimates. First, consistent with our knowledge of the BCEF's forest canopy, there is strong but localized spatial structure, which is reflected in the small noise to signal ratio α and short spatial range of ∼2 km $(-\log(0.05)/1.53)$. Second, as shown by its regression parameter estimate, β_{TC}, the tree cover predictor explains some variability in canopy height. Third, compared with the empirical semivariogram estimate of σ^2, which is ∼0.6 in Figure 4, the NNGP and SLGP estimate of 3.76 seems quite large. However, this difference in spatial variance is simply due to the objective function used to estimate the parameters. The parameters in the non-spatial regression from which the residuals were derived for the semivariogram were estimated to minimize the root mean squared difference between the observations and the fitted

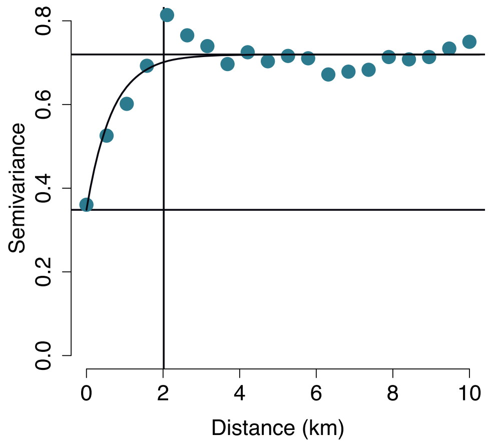
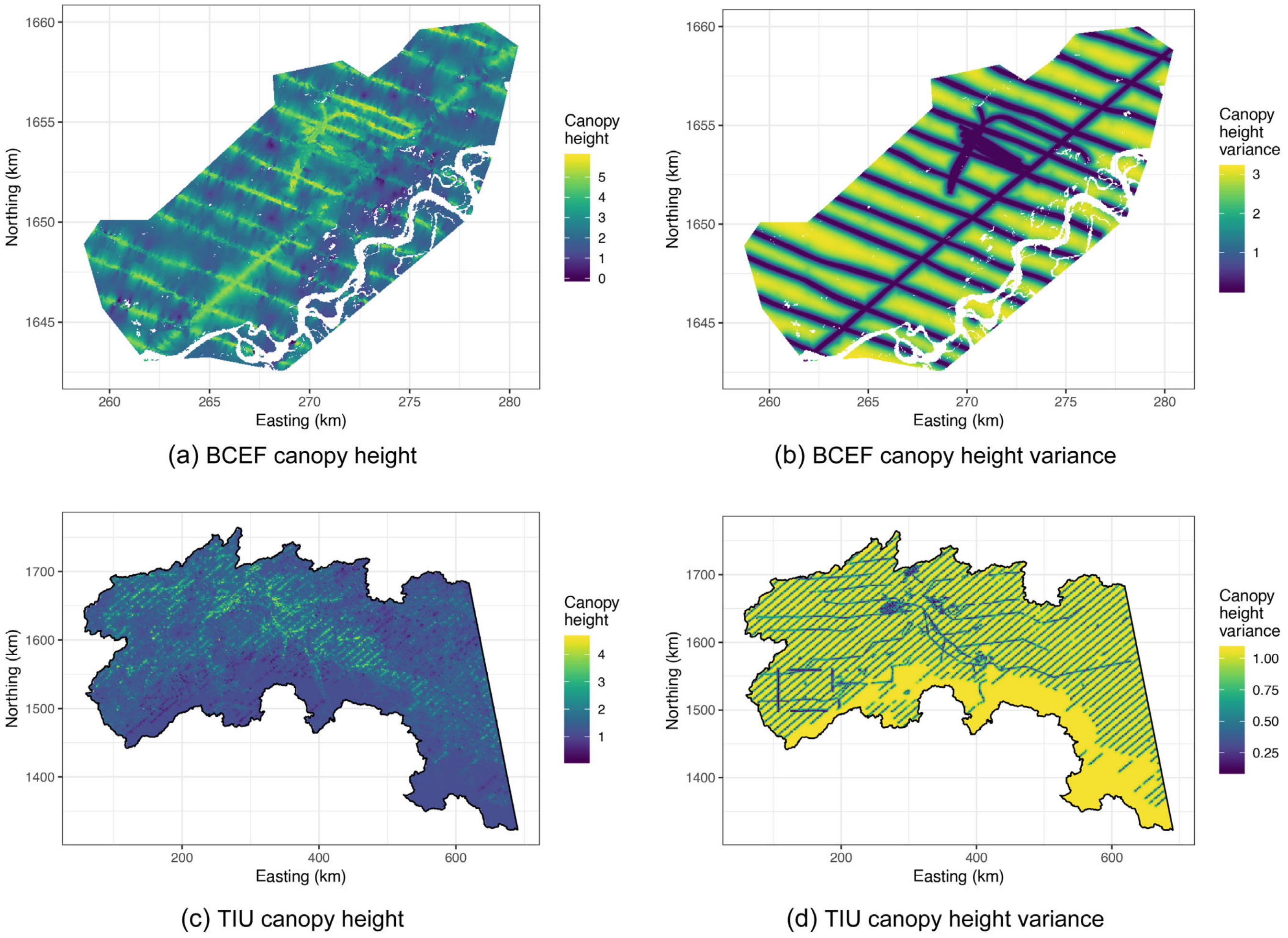


FIGURE 4 Semivariogram of BCEF nonspatial regression model residuals. Exponential covariance function estimate denoted by the curved line with associated estimates for τ^2, σ^2, and the effective spatial range are given by the lower horizontal, upper horizontal, and vertical lines respectively.



(a) BCEF canopy height

(b) BCEF canopy height variance

(c) TIU canopy height

(d) TIU canopy height variance

FIGURE 5 BCEF and TIU pixel-level SLGP model predicted square root of forest canopy height mean and associated variance

TABLE 3 TIU holdout set cross-validation CRPS-based parameter estimates and prediction metrics. Parameter variances estimates given in parentheses

			SLGP	
	NS	NNGP	$r = 196$	$r = 446$
β_0	0.55 (1.1[a])	1.0 (1.3[a])	0.98 (2.0[a])	1.0 (1.2[a])
β_{TC}	0.027 (2.1[b])	0.016 (1.0[b])	0.017 (1.0[b])	0.016 (1.0[b])
β_{Fire}	−0.03 (9.9[c])	0.12 (1.0[b])	0.12 (7.8[b])	0.12 (1.0[b])
α	–	0.13	0.255	0.13
τ^2	0.69	0.17	0.23	0.17
σ^2	–	1.31 (7.0[a])	0.61 (1.0[c])	1.31 (7.0[c])
ϕ	–	0.6	0.825	0.6
CRPS	0.47	0.38	0.38	0.38
RMSPE	0.83	0.69	0.69	0.69

Note: RMSPE values were generated from holdout set cross-validation RMSPE-based parameter estimates that are not reported in this table.

[a] Times 10^{-3}.

[b] Times 10^{-8}.

[c] Times 10^{-5}.

values. In contrast the parameter estimates in Table 2 were effectively chosen to minimize CRPS via cross-validation. Alternately, one could choose to use minimum RMSPE as the objective function for NNGP and SLGP parameterization (the result of which is given in the last row of Table 2), in which case the two models estimate $\sigma^2 = 0.8$ and $\tau^2 = 0.6$, which are much closer to the semivariogram estimates.

Figure 5a,b shows the predicted outcome and associated variance for every pixel across the BCEF. These prediction mean and uncertainty surfaces exhibit artifacts from the LiDAR sampling design. Notably, the mean shows more local patterns in canopy height where data is plentiful, and a retreat to the mean canopy height in regions far from observed data. The data sampling design and fairly short effective spatial range also exacerbate a feature of the NNGP and SLGP prediction algorithm. Specifically, because each prediction is informed by a set of nearest geographical neighbors, which often come from the same area along a LiDAR transect, the prediction surface has a *smudged* appearance perpendicular to the transects. Compared to the NNGP, the addition of the knots in the SLGP does smooth this artifact to some degree. The uncertainty surface Figure 5d, clearly shows observations along and adjacent to LiDAR transects inform prediction as reflected in the substantially higher precision.

We followed the same steps for the TIU analysis. Parameter estimates and prediction performance metrics are given in Table 3. Here, like the BCEF there is no notable difference between the NNGP and SLGP models in regard to their predictive performance. Both the percent tree cover and forest fire occurrence predictors explain some variability in canopy height as reflected by β_{TC}'s and β_{Fire}'s nonzero mean and small variance. The addition of the spatial random effects do improve prediction over that of the non-spatial regression. Also, we see both spatial models identify a slightly longer spatial range, that is, ~5 km but a large noise to signal variance ratio compared with the BCEF.

Finally, we fit the NNGP and SLGP with 200 knot models using $\alpha = 0.13$ and $\phi = 0.6$ to the entirety of the TIU's $n = 17{,}357{,}816$ observations. Using the parallelized model code and 18 CPU cores of the computer described in Section 3, run time for the NNGP model was ~9 and ~22 s for the SLGP model. Both models were used to predict canopy height for the TIU domain. Similar to the validation results presented in Table 3, the resulting TIU predictions were indistinguishable. Figure 5c,d shows the SLGP model's pixel-level mean and variance predictions for the TIU. All of the same prediction features seen in the BCEF are apparent in the TIU forest canopy.

6 | SUMMARY AND FUTURE WORK

This work further develops Bayesian NNGP models proposed by Datta, Banerjee, Finley, & Gelfand, 2016 with the aim to improve approximation, predictive performance, and most importantly computational efficiency to facilitate analysis of large data sets encountered in remote sensing applications. Massive computational gains are realized by using a conjugate method in place of MCMC-based inference for NNGP model parameter and predictive inference Finley et al. (2019). Extending recent work by Zhang, Sang, et al. (2019) we propose, implement, and illustrate a conjugate sparse plus low rank approximation (SLGP) that combines a Gaussian predictive process and nearest neighbor Gaussian process model. The proposed NNGP and SLGP algorithms and software implementation takes advantage of parallel computing and thrifty memory management to deliver Bayesian kriging inference for an unprecedentedly large data set (TIU $n \approx 17$ million locations) in less than a minute. The resulting forest canopy height models with associated pixel-level uncertainty quantification will help inform forest biomass and carbon models as part of an interior Alaska NASA Carbon Monitoring System. The level of computational efficiency delivered by conjugate NNGP and SLGP models opens many opportunities for previously unavailable statistically valid uncertainty assessment in remote sensing applications.

While full posterior inference via MCMC is often preferable, it is not yet computationally feasible for massive data sets. Also, from a practical standpoint, our experience and results presented here suggest that prediction accuracy and precision often do not suffer from fixing some spatial covariance parameters at reasonable values (e.g., the spatial decay parameter ϕ and noise to signal variance ratio α), hence we see value in pursuing computationally efficient NNGP and SLGP conjugate models such as those explored here. Following Zhang, Datta, et al. (2019), who detail how latent process inference can also be achieved for NNGP models within a conjugate Bayesian framework, our future work will focus upon explicitly estimating latent effects for SLGP models without using MCMC or other iterative algorithms such as INLA (Rue et al., 2009) or variational Bayes (Ren et al., 2011). Inference on the latent process (e.g., maps) can prove useful for identifying patterns in model residual and hypothesis testing.

Conceptually, multivariate and space-time versions can be easily adapted to be accommodated within the SLGP framework building upon the literature on low-rank dynamic spatial-temporal processes (Calder, 2008; Finley et al., 2012) and dynamic nearest-neighbor spatial-temporal processes (Datta, Banerjee, Finley, Hamm, & Schaap, 2016). Other avenues of research can focus on exploring conjugate NNGP and SLGP models for response and latent models (Zhang et al., 2021), for high-dimensional multivariate outcomes using spatial factor models (Ren & Banerjee, 2013; Taylor-Rodriguez et al., 2019; Zhang & Banerjee, 2022) and graphical Gaussian processes (Dey et al., 2021). Finally, as the use of scalable models and computational methods in complex data-driven environmental applications continue to evolve (Forlani et al., 2020; Tagle et al., 2020), it will be relevant to devise conjugate models and exact distribution theory to deliver inference without resorting to iterative algorithms.

ACKNOWLEDGMENTS

The work of Shinichiro Shirota and Sudipto Banerjee was supported, in part, by federal grants NSF/DMS-1513654, NSF/IIS-1562303 and NIH/NIEHS-1R01ES027027. Sudipto Banerjee was also supported by NSF/DMS-2113778. Andrew O. Finley was supported by NSF/DMS-1916395, NSF/EF-1253225, NASA Carbon Monitoring System, and USDA Forest Service, Region 9, Forest Health Protection, Northern Research Station.

ORCID

Sudipto Banerjee https://orcid.org/0000-0002-2239-208X

REFERENCES

Abdalati, W., Zwally, H., Bindschadler, R., Csatho, B., Farrell, S., Fricker, H., Harding, D., Dand Kwok, R., Lefsky, M., Markus, T., Marshak, A., Neumann, T., Palm, S., Schutz, B., Smith, B., Spinhirne, J., & Webb, C. (2010). The ICESat-2 laser altimetry mission. *Proceedings of the IEEE*, *98*(5), 735–751.

AICC (2016). Alaska Fire History Perimeter Polygons. U.S. Department of the Interior, Bureau of Land Management (BLM), Alaska Fire Service. https://fire.ak.blm.gov/predsvcs/maps.php.

Baccini, A., Friedl, M. A., Woodcock, C. E., & Warbington, R. (2004). Forest biomass estimation over regional scales using multisource data. *Geophysical Research Letters*, *31*, L10501. https://doi.org/10.1029/2004GL019782

Banerjee, S. (2017). High-dimensional Bayesian geostatistics. *Bayesian*. *Analysis*, *12*, 583–614.

Banerjee, S., Carlin, B. P., & Gelfand, A. E. (2014). *Hierarchical Modeling and Analysis for Spatial Data* (2nd ed.). Chapman and Hall/CRC.

Banerjee, S., Gelfand, A. E., Finley, A. O., & Sang, H. (2008). Gaussain predictive process models for large spatial data sets. *Journal of the Royal Statistical Society, Series B*, *70*, 825–848.

Calder, C. A. (2008). A dynamic process convolution approach to modeling ambient particulate matter concentrations. *Environmetrics*, *19*(1), 39–48.

Cook, B., Corp, L., Nelson, R., Middleton, E., Morton, D., McCorkel, J., Masek, J., Ranson, K., Ly, V., & Montesano, P. (2013). NASA Goddard's LiDAR, Hyperspectral and Thermal (G-LiHT) Airborne Imager. *Remote Sensing*, *5*(8), 4045–4066.

Dagum, L., & Menon, R. (1998). OpenMP: An industry standard API for shared-memory programming. *Computational Science & Engineering, IEEE*, *5*(1), 46–55.

Datta, A., Banerjee, S., Finley, A. O., & Gelfand, A. E. (2016). Hierarchical nearest-neighbor Gaussian process models for large geostatistical datasets. *Journal of the American Statistical Association*, *111*, 800–812.

Datta, A., Banerjee, S., Finley, A. O., Hamm, N. A. S., & Schaap, M. (2016). 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.

Dey, D., Datta, A., & Banerjee, S. (2021). Graphical Gaussian process models for highly multivariate spatial data. *Biometrika*, asab061. https://doi.org/10.1093/biomet/asab061

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

Finley, A. O., Banerjee, S., & Gelfand, A. E. (2012). Bayesian dynamic modeling for large space-time datasets using Gaussian predictive processes. *Journal of Geographical Systems*, *14*, 29–47.

Finley, A. O., Datta, A., Cook, B. C., Morton, D. C., Andersen, H. E., & Banerjee, S. (2019). Efficient algorithms for Bayesian nearest neighbor Gaussian processes. *Journal of Computational and Graphical Statistics*, *28*(2), 401-414. https://doi.org/10.1080/10618600.2018.1537924

Finney, M. A. (2004). *FARSITE: Fire area simulator - model development and evaluation. Technical Report Research Paper RMRS-RP-4, U.S. Department of Agriculture*. Rocky Mountain Research Station.

Forlani, C., Bhatt, S., Cameletti, M., Krainski, E., & Blangiardo, M. (2020). A joint Bayesian space–time model to integrate spatially misaligned air pollution data in R-INLA. *Environmetrics*, *31*(8), e2644.

GEDI (2014). Global ecosystem dynamics investigation LiDAR. Accessed May 1, 2015. http://science.nasa.gov/missions/gedi/.

G-LiHT (2019) Goddard's LiDAR, H.T.I. Accessed November 5, 2019. https://gliht.gsfc.nasa.gov/

Gneiting, T., & Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, *102*, 359–378.

Guinness, J. (2018). Permutation and grouping methods for sharpening Gaussian process approximations. *Technometrics*, *60*(4), 415-429. https://doi.org/10.1080/00401706.2018.1437476

Hansen, M. C., Potapov, P. V., Moore, R., Hancher, M., Turubanova, S. A., Tyukavina, A., Thau, D., Stehman, S. V., Goetz, S. J., Loveland, T. R., Kommareddy, A., Egorov, A., Chini, L., Justice, C. O., & Townshend, J. R. G. (2013). High-resolution global maps of 21st-century forest cover change. *Science*, *342*(6160), 850–853.

Heaton, M., Datta, A., Finley, A., Furrer, R., Guhaniyogi, R., Gerber, F., Gramacy, R. B., Guiness, J., Hammerling, D., Katzfuss, M., Lindgren, F., Nychka, D., Sun, F., & Zammit-Mangion, A. (2019). A case study competition among methods for analyzing large spatial data. *Journal of Agricultural Biological and Environmental Statistics,* 398–425. https://doi.org/10.1007/s13253-018-00348-w

Hurtt, G. C., Dubayah, R., Drake, J., Moorcroft, P. R., Pacala, S. W., Blair, J. B., & Fearon, M. G. (2004). Beyond potential vegetation: Combining lidar data and a height-structured model for carbon studies. *Ecological Applications*, *14*(3), 873–883.

ICESat-2 (2015). Ice, cloud, and land elevation satellite-2. Accessed May 1, 2015. http://icesat.gsfc.nasa.gov/.

Katzfuss, M. (2017). A multi-resolution approximation for massive spatial datasets. *Journal of the American Statistical Association*, *112*, 201–214.

Katzfuss, M., & Guinness, J. (2021). A General Framework for Vecchia Approximations of Gaussian Processes. *Statistical Science*, *36*(1), 124–141.

Klein, T., Randin, C., & Korner, C. (2015). Water availability predicts forest canopy height at the global scale. *Ecology Letters*, *18*(12), 1311–1320.

Lefsky, M. A. (2010). A global forest canopy height map from the moderate resolution imaging spectroradiometer and the geoscience laser altimeter system. *Geophysical Research Letters*, *37*(15), L15401.

Liu, J. S., Wong, W. H., & Kong, A. (1994). Covariance structure of the Gibbs sampler with applications to the comparisons of estimators and augmentation schemes. *Biometrika*, *81*, 27–40.

LTER. Bonanza Creek experimental forest. Accessed May 11, 2019. http://www.lter.uaf.edu/research/study-sites-bcef.

Ma, P. & Kang, E. L. (2017). Fused Gaussian process for very large spatial data. arXiv:1702.08797.

Ren, Q., & Banerjee, S. (2013). Hierarchical factor models for large spatially misaligned data: a low-rank predictive process approach. *Biometrics*, *69*, 19–30.

Ren, Q., Banerjee, S., Finley, A. O., & Hodges, J. S. (2011). Variational Bayesian methods for spatial data analysis. *Computational Statistics and Data Analysis*, *55*(12), 3197–3217.

Rue, H., Martino, S., & Chopin, N. (2009). Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the Royal Statistical Society, Series B*, *71*, 319–392.

Sang, H., & 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.

Sang, H., Jun, M., & Huang, J. Z. (2011). Covariance approximation for large multivariate spatial data sets with an application to multiple climate model errors. *Annals of Applied Statistics*, *4*, 2519–2548.

Simard, M., Pinto, N., Fisher, J. B., & Baccini, A. (2011). Mapping forest canopy height globally with spaceborne lidar. *Journal of Geophysical Research. Biogeosciences*, *116*, G04021. https://doi.org/10.1029/2011JG001708

Stratton, R. D. (2006). *Guidance on spatial wildland fire analysis: models, tools, and techniques*. Technical Report General Technical Report RMRS-GTR-183, U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station.

Sun, Y., Li, B., & Genton, M. (2012). *Geostatistics for large datasets*. In J. Montero, E. Porcu, & M. Schlather (Eds.), *Advances and Challenges in Space-time Modelling of Natural Events* (pp. 55–77). Springer-Verlag.

Tagle, F., Genton, M. G., Yip, A., Mostamandi, S., Stenchikov, G., & Castruccio, S. (2020). A high-resolution bilevel skew-*t* stochastic generator for assessing Saudi Arabia's wind energy resources. *Environmetrics*, *31*(7), e2628.

Taylor-Rodriguez, D., Finley, A. O., Datta, A., Babcock, C., Andersen, H. E., Cook, B. C., Morton, D. C., & Banerjee, S. (2019). Spatial factor models for high-dimensional and large spatial data: an application in forest variable mapping. *Statistica Sinica*, *29*, 1155–1180.

Vecchia, A. V. (1988). Estimation of model identification for continuous spatial processes. *Journal of the Royal Statistical Society, Series B*, *50*, 297–312.

Zhang, B., Sang, H., & Huang, J. Z. (2019). Smoothed full-scale approximation of Gaussian process models for computation of large spatial datasets. *Statistica Sinica*, 29, 1711-1737. https://doi.org/10.5705/ss.202017.0008

Zhang, L., & Banerjee, S. (2022). Spatial factor modeling: A Bayesian matrix-normal approach for misaligned data. *Biometrics*, *78*(2), 560–573.

Zhang, L., Banerjee, S., & Finley, A. O. (2021). High-dimensional multivariate geostatistics: A Bayesian matrix-normal approach. *Environmetrics*, *32*(4), e2675.

Zhang, L., Datta, A., & Banerjee, S. (2019). Practical Bayesian modeling and inference for massive spatial data sets on modest computing environments. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, *12*(3), 197–209.

Zhang, X. (2016). An optimized BLAS library based on gotoblas2. Accessed June 1, 2015. https://github.com/xianyi/OpenBLAS/.

How to cite this article: Shirota, S., Finley, A. O., Cook, B. D., & Banerjee, S. (2023). Conjugate sparse plus low rank models for efficient Bayesian interpolation of large spatial data. *Environmetrics*, *34*(1), e2748. https://doi.org/10.1002/env.2748

APPENDIX

We present the algorithms discussed in Section 3 for rendering posterior samples from the conjugate SLGP model (Algorithm 1).

Algorithm 1. Hyper parameter tuning

```
1. Fix $\alpha$ and $\phi$, split the data into $K$ folds.
   (a) Find the collection of neighbor sets $\mathcal{N} = \{N(i,k) : i = 1, 2, \dots, n; k = 1, 2, \dots, K\}$
   (b) Construct $\mathbf{\Omega}[S(-k), S(-k)]$, $\mathbf{J}[S(-k), S^*]$ and $\mathbf{X}^* = (\mathbf{X}, \mathbf{J}[S(-k), S^*])$ for $k = 1, 2, \dots, K$ In practice, we just calculate
       $\mathbf{\Omega}[S, S]$, $\mathbf{J}[S, S^*]$ one time for each $(\alpha, \phi)$. Then extract $\mathbf{\Omega}[S(-k), S(-k)]$, $\mathbf{J}[S(-k), S^*]$ from them.
2. Obtain posterior means for $\boldsymbol{\beta}$ and $\sigma^2$ after removing the $k$th fold of the data:
   (a) Obtain A(k) and D(k) from $\mathbf{\Omega}[S(-k), S(-k)]$
   (b) F=solve(V$_{\beta^*}$,I); f=solve(V$_{\beta^*}$, $\boldsymbol{\mu}_{\beta^*}$) In practice this step is calculated one time for each $(\alpha, \phi)$
   (c) Solve for $(p+r) \times (p+r)$ matrix $\mathbf{B}(k)$ and $(p+r) \times 1$ vector $\mathbf{b}(k)$
       for(i in 1:p+r){
         b(k)[i] = f[i]
           + qf(X$^*$[S(-k),i],y[S(-k)],A(k),D(k))
         for(j in 1:p+r) {
           B(k)[i,j] = qf(X$^*$[S(-k),i],X$^*$[S(-k),j],
             A(k),D(k)) + F[i,j]
         }
       }
   (d) Obtain V(k), g(k), a$^*_\sigma$(k), b$^*_\sigma$(k), by the same way in the step 2.(d) of algorithm 5 in Finley et al. (2018)
   (e) $\hat{\beta}^*$ = g(k), $\hat{\sigma}^2$ =b$^*_\sigma$(k)/(a$^*_\sigma$(k)-1)
3. Predicting posterior means of $\mathbf{y}(S[k])$:
   for(s in S(k)){
     N(s,k) = m-nearest neighbors of s from S(-k)
     z = Omega(s, N(s,k))
     w = solve(Omega[N(s,k),N(s,k)],z)
     $\hat{y}(s)$ = dot(x$^*$(s), g(k)) + dot(w,
       (y[N(s,k)] - dot(X$^*$[N(s,k),],g(k))))
     v0 = dot(u, gemv(V(k),u)) + 1 + alpha
       - dot(w,z)
     Var($\hat{y}$(s)) = b$^*$(k)v0/(a$^*$(k)-1)
   }
4. Compute a scoring rule over $K$ folds
5. Cross validation for choosing $\alpha$ and $\phi$:
   (a) Repeat steps (2) and (3) for $G$ values of $\alpha$ and $\phi$
   (b) Choose $\alpha_0$ and $\phi_0$ as the value that minimize the average scoring rule
   Algorithm: Parameter estimation and prediction from conjugate SLGP model
6. Repeat step 2 with $(\alpha_0, \phi_0)$ and the full data to get $(\boldsymbol{\beta}^*, \sigma^2)|\mathbf{y}$
7. Repeat step 3 with $(\alpha_0, \phi_0)$ and the full data to predict at a new location $\mathbf{s}_0$ to obtain the mean and variance of $y(\mathbf{s}_0)|\mathbf{y}$.
```