A Scalable Partitioned Approach to Model Massive Nonstationary Non-Gaussian Spatial Datasets

Benjamin Seiyon Lee and Jaewoo Park

Department of Statistics, George Mason University, Fairfax, VA; Department of Statistics and Data Science, Yonsei University, Seoul, Republic of Korea; Department of Applied Statistics, Yonsei University, Seoul, Republic of Korea

ABSTRACT
Nonstationary non-Gaussian spatial data are common in many disciplines, including climate science, ecology, epidemiology, and social sciences. Examples include count data on disease incidence and binary satellite data on cloud mask (cloud/no-cloud). Modeling such datasets as stationary spatial processes can be unrealistic since they are collected over large heterogeneous domains (i.e., spatial behavior differs across subregions). Although several approaches have been developed for nonstationary spatial models, these have focused primarily on Gaussian responses. In addition, fitting nonstationary models for large non-Gaussian datasets is computationally prohibitive. To address these challenges, we propose a scalable algorithm for modeling such data by leveraging parallel computing in modern high-performance computing systems. We partition the spatial domain into disjoint subregions and fit locally nonstationary models using a carefully curated set of spatial basis functions. Then, we combine the local processes using a novel neighbor-based weighting scheme. Our approach scales well to massive datasets (e.g., 2.7 million samples) and can be implemented in nimble, a popular software environment for Bayesian hierarchical modeling. We demonstrate our method to simulated examples and two massive real-world datasets acquired through remote sensing.

1. Introduction
Nonstationary spatial models have been used in a wide range of scientific applications, including remote sensing (Heaton, Christensen, and Terres 2017), precision agriculture (Katzfuss 2013), precipitation modeling (Risser and Calder 2015), and air pollutant monitoring (Fuentes 2002). Simple models assume second-order stationarity of the spatial process; however, this can be unrealistic since data are collected over heterogeneous domains. Here, the spatial processes can exhibit localized spatial behaviors. Although several methods (see Fuentes 2002; Risser and Calder 2015; Heaton, Christensen, and Terres 2017) have been developed for modeling nonstationary spatial data, these have focused primarily on Gaussian spatial data. Moreover, fitting these models poses both computational and inferential challenges, especially for large datasets. In this article, we propose a scalable algorithm for fitting nonstationary non-Gaussian datasets. Our approach captures nonstationarity by partitioning the spatial domain and modeling the localized spatial processes using basis expansions. This new algorithm is computationally efficient in that: (a) partitioning the spatial domain permits parallelized computation on high-performance computing systems; and (b) basis approximation of spatial processes dramatically reduces the computational overhead.

There is a growing literature on modeling nonstationary spatial datasets. Weighted average methods (Fuentes 2001; Kim, Mallick, and Holmes 2005; Risser and Calder 2015) combine localized spatial models to reduce computational costs. Basis function approximations (Nychka, Wikle, and Royle 2002; Katzfuss 2013, 2017; Hefley et al. 2017) represent complex spatial processes using linear combinations of spatial basis functions. Higdon (1998) and Paciorek and Schervish (2006) represent a nonstationary process using convolutions of spatially varying kernel functions. Based on the spatial partitioning strategies, some of these approaches are amenable to massive spatial datasets. For example, Heaton, Christensen, and Terres (2017) develops a computationally efficient approach for large nonstationary spatial data by partitioning an entire domain into disjoint sets using a hierarchical clustering algorithm. Katzfuss (2017) constructs basis functions at multiple resolutions based on recursive partitioning of the spatial region. Guhriyogi and Banerjee (2018) proposes a divide-and-conquer approach to generate a global posterior distribution by combining local posterior distributions from each subsample. Though these methods scale well to large datasets, they are limited to Gaussian responses.

Spatial generalized linear mixed models (SGLMMs) (Diggle, Tawn, and Moyeed 1998) are a popular class of models designed for non-Gaussian spatial datasets. SGLMMs are widely used for both areal and point-referenced data, where latent Gaussian random fields can account for the spatial correlations. However, fitting SGLMMs for massive spatial datasets is demanding since the dimension of the correlated spatial processes grows...
with respect to the number of observations. Although several computational methods (Banerjee et al. 2008; Guan and Haran 2018; Lee and Haran 2021) have been proposed for large non-Gaussian spatial datasets, these methods assume second-order stationarity of the latent spatial processes. Recently, Zilber and Katzfuss (2021) propose a Vecchia-Laplace approximation for big non-Gaussian spatial data. Although the approach can be extended to the nonstationary case by adopting nonstationary covariance functions, they focus on stationary covariance functions in the illustrative examples. Furthermore, this work has not been applied to massive spatial datasets (e.g., millions of observations) that we consider in this article.

In this article, we propose a scalable approach for modeling massive nonstationary non-Gaussian spatial datasets. Our smooth mosaic basis approximation for nonstationary SGLMMs (SMB-SGLMMs) combines key ideas from weighted average approaches and basis approximations. SMB-SGLMM consists of four steps: (a) partition the spatial region using a clustering algorithm (Heaton, Christensen, and Terres 2017); (b) generate localized spatial basis functions; (c) fit nonstationary basis-representation models to each partition; and (d) smooth the local processes using distance-based weighting scheme (smooth mosaic). Due to partitioning and localized model fitting, we can leverage parallel computing, which greatly increases the scalability of the SMB-SGLMM method. The idea of the proposed algorithm is closely related to the Bayesian committee machine (Tresp 2000) that combines the predictions from the local models with a weighting scheme. The Bayesian committee machine scales linearly in training data; therefore, such divide-and-conquer approaches scale well for massive datasets. Our method provides an automated mechanism for selecting appropriate spatial basis functions. We also provide ready-to-use code written in nimble (de Valpine et al. 2017), a software environment for Bayesian inference.

The outline for this article is as follows. In Section 2, we introduce several nonstationary modeling approaches. We discuss the extension of stationary SGLMMs to nonstationary SGLMMs and the inherent challenges. In Section 3, we propose SMB-SGLMMs for massive spatial data and provide implementation details. Furthermore, we investigate the computational complexity of our method in detail. In Section 4, we study the performance of SMB-SGLMMs through simulated data examples. In Section 5, we apply SMB-SGLMMs to massive remote sensing datasets. We conclude with a discussion in Section 6.

2. Nonstationary Modeling for Spatial Data

Let \( Z = \{Z(s_i)\}_{i=1}^{N} \) be the observed data and \( X \in \mathbb{R}^{N \times p} \) be the matrix of covariates at the spatial locations \( s_i = \{s_i\}_{i=1}^{N} \) in a spatial domain \( S \subseteq \mathbb{R}^2 \). \( W = \{W(s_i)\}_{i=1}^{N} \) is a mean-zero Gaussian process with covariance matrix \( \Sigma \in \mathbb{R}^{N \times N} \). Then SGLMMs can be defined as

\[
g(\mathbb{E}[Z|\beta, W]) = \eta = X\beta + W \quad W \sim N(0, \Sigma)
\]

with link function \( g(\cdot) \) and linear predictor \( \eta \). Standard SGLMMs (Diggle, Tawn, and Moyeed 1998) consider a second-order stationary Gaussian process for \( W \) for their convenient mathematical framework. However, this assumption can be unrealistic for spatial processes existing in large heterogeneous domains (see Bradley, Cressie, and Shi (2016), for a discussion). A natural extension to (1) is to model \( W \) as a nonstationary spatial process. There is an extensive literature on modeling nonstationary spatial data (Sampson 2010) such as: (a) weighted-average methods, (b) basis function methods, and (c) process convolutions. Our method is motivated by these nonstationary modeling approaches.

Weighted average methods (Fuentes 2001) divide the spatial region \( S \) into disjoint partitions and fit locally stationary models to each partition. For example, Kim, Mallick, and Holmes (2005) and Heaton, Christensen, and Terres (2017) partition the spatial domain through Voronoi tessellation. Then, the global process is constructed by combining the locally stationary processes via a weighted average. The weights are computed using the distances between the observation locations and “center” of the localized processes. These approaches scale well by taking advantage of parallel computation (see Risser and Calder 2015; Heaton, Christensen, and Terres 2017).

Basis functions approach represent the nonstationary covariance structure as an expansion of spatial basis functions \( \{\Phi_j(s)\}_{j=1}^{m} \). Let \( \Phi \) be an \( m \times m \) matrix with columns indicate the basis functions and rows indicate locations \( \Phi_j(s_i) \). Then we can construct a nonstationary spatial process as \( W \equiv \Phi \delta and \delta \sim N(0, \Sigma_{\phi}) \) where \( \delta \) is the coefficients of basis functions. We approximate the covariance structure as \( \Phi \Sigma_{\phi} \Phi^T \), which is not dependent solely on the lag between locations; hence, this is nonstationary. Different types of basis functions have been used, for instance eigenfunctions obtained from the empirical covariance (Holland et al. 1999), multiresolution basis functions (Nychka, Wikle, and Royle 2002; Nychka et al. 2015; Katzfuss 2017), and computationally efficient low-rank representation of nonstationary covariance (Katzfuss 2013).

Process convolutions represent the nonstationary spatial processes through convolutions of spatially varying kernel and Brownian motion. For an arbitrary \( s \in S \), \( W(s) = \int_{S} K(u) dW(u) \) where \( K(u) \) is a kernel function centered at location \( s \) and \( W(\cdot) \) is a Brownian motion. Higdon (1998) use Gaussian kernels under this framework. Several extensions have been proposed including creating closed-form nonstationary Matérn covariance functions (Paciorek and Schervish 2006), extension to multivariate spatial process (Kleiber and Nychka 2012), and efficient local likelihood approaches (Risser and Calder 2015).

We note that these nonstationary models have focused on Gaussian responses. Direct application of these methods to (1) is challenging because we cannot obtain closed-form maximum likelihood estimates by marginalizing out \( W \). Within the Bayesian framework, updating conditional posterior distributions requires a computational complexity of \( O(N^3) \), which becomes infeasible even for moderately large size datasets (e.g., binary satellite data with 100,000 observations). Although efficient projection-based approaches (see Banerjee et al. 2008; Guan and Haran 2018; Lee and Haran 2021) have been developed for non-Gaussian hierarchical spatial models, they are assuming stationarity of \( W \). There is a growing literature on modeling nonstationarity non-Gaussian spatial data. The integrated nested Laplace approximation (INLA) (Rue, Martino, 2018)
and Chopin 2009; Lindgren, Rue, and Lindström 2011) accounts for nonstationarity by modifying the stochastic partial differential equation with spatially varying covariance parameters (Ingebrigtsen, Lindgren, and Steinsland 2014); the resulting solutions are nonstationary Gaussian random fields (see also a comparative review of INLA in Bakka et al. 2018). Bradley, Holan, and Wikle (2020) introduce a family of novel latent conjugate multivariate (LCM) distributions which makes it possible to conduct Bayesian inference for large dependent non-Gaussian data. LCM permits fast sampling from the full conditional distributions and accounts for nonstationarity by using a basis representation of the random field. Bradley, Holan, and Wikle (2018) propose a special case of LCM for high-dimensional spatio-temporal count data using a multivariate log-gamma distribution. Sengupta and Cressie (2013), Sengupta et al. (2016), and Shi and Kang (2017) model large non-Gaussian spatial datasets by representing the latent spatial field via nested bi-square basis functions at varying resolutions. Basis representation approaches naturally generate nonstationary covariance structures and can achieve dimension reduction, depending on the number of basis functions.

By using some of these key ideas, we developed a class of partitioned nonstationary models for non-Gaussian spatial data. By partitioning the spatial domain, we can model the local spatial processes with parallel computation. With relatively little effort, scalable computation for Bayesian hierarchical models is possible, and we can further reduce computing time by increasing the number of available cores. Our method is computationally efficient and provides accurate predictions over large heterogeneous spatial domains.

3. Smooth Mosaic Basis Approximation for Nonstationary SGLMMs

We propose a smooth mosaic basis approximation for nonstationary SGLMMs designed for massive spatial datasets. Our method is outlined below and illustrated in Figure 1:

Step 1. Partition the spatial domain into disjoint subregions.
Step 2. Construct data-driven basis functions for each subregion.
Step 3. Fit a locally nonstationary basis function model to each subregion in parallel.
Step 4. Construct the global nonstationary process as a weighted average of local processes.

3.1. Partitioned Nonstationary Spatial Models

Step 1. Partition the spatial domain into disjoint subregions
We use an agglomerative clustering approach (Heaton, Christensen, and Terres 2017) to partition the spatial domain \( S \) into \( K \) subregions \( \{S_k\}_{k=1}^K \), which satisfy \( \bigcup_{k=1}^K S_k = S \). We fit a nonspatial generalized linear model (\texttt{glm} function in R) using responses \( Z \) and covariates \( X \). Then we obtain the spatially correlated residuals \( \{\epsilon(s_i)\}_{i=1}^N \). For all \( i \neq j \), we calculate the dissimilarity between \( \epsilon(s_i) \) and \( \epsilon(s_j) \) as \( d_{ij} = |\epsilon(s_i) - \epsilon(s_j)|/\|s_i - s_j\| \) from spatial finite differences (Banerjee and Gelfand 2006).

Figure 1. Illustration for the partitioned nonstationary approach for simulated \( W \). (a) Nonstationary \( W \) is partitioned through 16 subregions; different colors indicate disjoint partitions. (b) For each partition, thin plate spline basis functions are constructed at knots; basis functions represent distinct spatial patterns. (c) The Local nonstationary model is fit to each partition using a linear combination of basis functions. (d) The global nonstationary process is obtained via a weighted average of the local processes.
Step 2. Construct data-driven basis functions for each subregion

For each $S_k$, we generate a collection of spatial basis functions. We have $Z_k = \{Z(s) : s \in S_k\} \in \mathbb{R}^{N_k}$, the observations belong to $S_k$ and $N = \sum_{k=1}^{K} N_k$. $X_k$ is an $N_k \times p$ matrix of covariates. Consider the knots (grid points) $(u_{kj})_{j=1}^{m_k}$ over $S_k$ ($m_k \ll N_k$). These knots can define spatial basis functions such as radial basis functions (Nychka et al. 2015) and eigenbasis functions (Banerjee et al. 2008). In this study, we consider thin plate splines defined as $\Phi_k(s) = \|s - u_{kj}\|^2 \log(\|s - u_{kj}\|)$ (Holland et al. 1999; Banerjee et al. 2013; Guan and Haran 2018), principal components (Higdon et al. 2008; Cressie 2015), and Moran’s basis (Hughes and Haran 2013).

Step 3. Fit a locally nonstationary basis function model to each subregion

For each $S_k$, we can represent the spatial random effects as $W_k \approx \Phi_k \delta_k$ and model the conditional mean $\mathbb{E}[Z_k | \beta_k, \Phi_k, \delta_k]$ as

$$g(\mathbb{E}[Z_k | \beta_k, \Phi_k, \delta_k]) := \eta_k = X_k \beta_k + \Phi_k \delta_k$$

$$\delta_k \sim N(0, \Sigma_{\Phi_k})$$

(2)

where $\Sigma_{\Phi_k}$ is a covariance of basis coefficients $\delta_k$. Here we set $\Sigma_{\Phi_k} = \sigma_k^2 I_k$, as in a discrete approximation of a nonstationary-Gaussian process (Higdon 1998). This basis representation approximates the covariance using a linear combination of basis functions $\sigma_k^2 \Phi_k \Phi_k^T$. By construction, the basis approximation can capture the nonstationary behavior of the spatial process (Figure 1(c)). Since we typically choose $m_k \ll N_k$, basis representations can drastically reduce computational costs by avoiding large matrix operations. We provide implementation details in Section 3.3. In addition, a clever choice of $\Phi_k$ can also reduce correlations in $\delta_k$, resulting in fast mixing MCMC algorithms (Haran, Hodges, and Carlin 2003). For an exponential family distribution $F(\cdot)$, the partition-specific hierarchical spatial model is as follows:

Data Model: $Z_k | \eta_k \sim f(\eta_k)$

$$g(\mathbb{E}[Z_k | \beta_k, \Phi_k, \delta_k]) := \eta_k = X_k \beta_k + \Phi_k \delta_k$$

(3)

Process Model: $\delta_k | \sigma_k^2 \sim N(0, \sigma_k^2 I_k)$

Parameter Model: $\beta_k \sim p(\beta_k), \sigma_k^2 \sim p(\sigma_k^2)$

We complete the hierarchical model by assigning prior distributions for $\beta_k$ and $\sigma_k^2$.

Step 4. Construct the global nonstationary process as a weighted average

To construct the global process, we use a weighted average of the fitted local processes. Note that $\Phi_k \in \mathbb{R}^{N\times m_k}$ is the basis functions matrix consisting of thin plate splines $\Phi_k(s) = \|s - u_{kj}\|^2 \log(\|s - u_{kj}\|)$ for $s \in S_k$, where $(u_{kj})_{j=1}^{m_k}$ are the knots over $S_k$. Here, we introduce another related basis functions matrix. We define $\widetilde{\Phi}_k \in \mathbb{R}^{N\times m_k}$ by evaluating $\Phi_k(s)$ for all locations $s \in S$. Let $\tilde{\Phi}_k(s) \in \mathbb{R}^{m_k}$ be the row of $\widetilde{\Phi}_k$ corresponding to spatial location $s \in S$. Since $W_k(s) \approx \tilde{\Phi}_k(s) \delta_k$, we have:

$$W(s) = \sum_{k=1}^{K} c_k(s) W_k(s) \approx \sum_{k=1}^{K} c_k(s) \tilde{\Phi}_k(s) \delta_k$$

(4)

$$c_k(s) = \frac{\exp \left(-\|s - \tilde{s}_k\|^2 / \|s - \tilde{s}_k\|^2 \right)}{\sum_{j=1}^{K} \exp \left(-\|s - \tilde{s}_j\|^2 / \|s - \tilde{s}_j\|^2 \right)}$$

if $\|s - \tilde{s}_j\| \leq \gamma$ otherwise 0, where $\tilde{s}_k \in S_k$ is the closest knot point to $s$. The weights are normalized such that $\sum_{k=1}^{K} c_k(s) = 1$. In (4), we use a modified version (Joseph and Kang 2011) of the inverse distance weighting (Shepard 1968) to avoid tuning the scale parameter in the Gaussian kernel; this can be useful when the spatial domain of interest is not scaled. This weighting scheme shares similarities with Risser and Calder (2015) in that the weight $c_k(s)$ is the negative exponential function of the distance between $s$ and $\tilde{s}_k$; hence, shorter distances result in higher weights. We assign a 0 weight if the distance exceeds a threshold, or weighting radius, $\gamma$. We present details about choice of $\gamma$ in Section 3.3. The weighted average of the local processes approximates the nonstationary global process (Figure 1(d)). Similarly, a global linear predictor $\eta(s)$ can be written as

$$\eta(s) = \sum_{k=1}^{K} c_k(s) [X(s)^T \beta_k + \tilde{\Phi}_k(s) \delta_k]$$

(5)

Here $X(s) \in \mathbb{R}^p$ is a vector of the covariate matrix $X$ for location $s$, $\beta_k \in \mathbb{R}^p$ is corresponding regression coefficients. Our method provides a partition-varying estimate of $\beta_k$. This is because the fixed effects may have spatially varying (nonstationary) behavior over large heterogeneous spatial domains. Therefore, as in Heaton, Christensen, and Terres (2017) we provide a partition-varying $\beta_k$ in our applications.

Divide and conquer strategies have been developed for scalable Bayesian computation (see Srivastava et al. 2015; Minsker et al. 2017; Campbell and Broderick 2019). Srivastava et al. (2015) and Minsker et al. (2017) approximate the full posterior as a weighted average of subset posteriors using the Wasserstein barycenter and geometric median, respectively. Campbell and Broderick (2019) use the Hilbert coreset to construct a sparse weighted log-likelihood that approximates the full log-likelihood. Compared to these approaches, our method uses distance-based weights (4) to combine locally nonstationary processes (see Fuentes 2001; Risser and Calder 2015; Li and Sun...
In spatial models, nonstationarity may exist in both the mean and covariance structures. For the mean structure, we model the regression coefficients as partition-varying constants \( \mathbf{\beta}_k \), which results in a locally stationary mean. If a locally nonstationary mean field is desired, one can embed spatially varying coefficients (Gelfand et al. 2003; Mu, Wang, and Wang 2018) (i.e., \( \mathbf{\beta}_k(s) \)) into our model (3). This generalization has a similar hierarchical structure to our original model, but now requires a basis approximation of \( \mathbf{\beta}_k(s) \), which can increase computational costs. In this study, we focus on locally stationary mean structures, while allowing for locally nonstationary covariance structures. We provide a detailed discussion for locally nonstationary models in Section 3.2.

**Spatial prediction**

Spatial prediction at unobserved locations is of great interest in many scientific applications. Let \( Z(s^*) \) be an unobserved response at an arbitrary location \( s^* \in S \). Using thin plate splines, we can construct localized basis functions as

\[
\Phi_{kj}(s^*) = \| s^* - u_{kj} \|^2 \log(\| s^* - u_{kj} \|),
\]

where we have \( \{ u_{kj} \}_{j=1}^{m_k} \) knots in partition \( k \). As in (5), the global prediction is

\[
\eta(s^*) = \sum_{k=1}^{K} c_k(s^*)^\top \mathbf{\beta}_k + \Phi_k^\top(s^*) \mathbf{\delta}_k. \tag{6}
\]

For given posterior samples \( \{ \mathbf{\beta}_k, \mathbf{\delta}_k \}_{k=1}^{K} \), we can obtain a posterior predictive distribution of \( \eta(s^*) \). Here, the likelihood function at an unobserved location \( s^* \) is defined through the exponential family distribution with a link function \( g \{ \mathbb{E}[Z(s^*)|\mathbf{\beta}_k, \Phi_k(s^*), \mathbf{\delta}_k] \} = \eta(s^*) \).

### 3.2. Locally Nonstationary Models

Paciorek and Schervish (2003), Stein (2005), and Paciorek and Schervish (2006) present a parametric family of nonstationary Matérn class covariance functions based on process convolutions where nonstationarity results from the spatially varying covariance parameter \( \mathbf{\theta}(s) \). However, it is challenging to estimate \( \mathbf{\theta}(s) \) in practice due to optimization and identifiability issues (Anderes and Stein 2011; Li and Sun 2019). To address such problems, locally stationary covariance functions have been used to model nonstationary spatial processes. For instance, Paciorek and Schervish (2006) and Anderes and Stein (2011) assume constant covariance parameters for each partition and fit a stationary model for each partition (i.e., \( \mathbf{\theta}(s) = \mathbf{\theta}_k \) for \( s \in S_k \)). By using distance-based weights, Risser and Calder (2015) and Fouedjio, Desassis, and Rivoirard (2016) provide smoothly weighted estimates of the covariance parameters. Fuentes (2001) also represents nonstationary process via a weighted average of local stationary process by assuming a constant Matérn covariance parameter for each partition. These local stationary models have advantages that they can represent complex nonstationary process through the interpretable stationary covariance functions from each partition.

Locally stationary models can be a practical option, but their performance is strongly affected by how we partition the spatial domain. Precisely specifying the locally stationary regions is a nontrivial task. Furthermore, we need enough observations within each partition to accurately estimate the local covariance structure (Li and Sun 2019). To obtain a suitable sample size, we may have to collect data over a larger spatial region, which is likely to exhibit locally nonstationary behavior; therefore, there is potential for model misspecification issues. A recent exception is Li and Sun (2019) which use higher-order polynomial approximations for \( \mathbf{\theta}(s) \) that is locally nonstationary; however, they only study Gaussian responses. SMB-SGLMMs also permit local nonstationarity within each subregion by representing a local process through thin plate splines (Step 3), but applies to non-Gaussian responses. Locally stationary models can be a special case depending on the choice of basis functions. For each partition \( S_k \), we can set \( \mathbf{\Phi}_k = \mathbf{U}_k(m) \mathbf{D}_k^{1/2} \), where \( \mathbf{U}_k(m) \) is a matrix of the leading \( m \) eigenvectors of a stationary Matérn class covariance function and \( \mathbf{D}_k^{1/2} \) is a diagonal matrix with corresponding eigenvalues.

Locally nonstationary models can be more robust against partition misspecification than locally stationary models. Let \( S^{(c)} \) be the set of partitions with correct specification; the true covariance functions under these partitions are assumed to be stationary Matérn covariance function with known parameters. Then we can define the set of incorrectly specified partitions \( S^{(i)} = S \setminus S^{(c)} \) (i.e., \( S^{(i)} \cup S^{(c)} = S \)), where the true covariance functions are nonstationary. For partitions \( S_k \in S^{(i)} \), the \( m \)-leading eigenbasis functions \( \mathbf{U}_k(m) \mathbf{D}_k^{1/2} \) can recover the true process because \( \mathbf{U}_k(m) \mathbf{D}_k(m) \mathbf{U}_k(m)^\top \) is the best rank- \( m \) approximation to stationary Matérn covariance (Stewart 1993; Banerjee et al. 2013) with respect to both spectral and Frobenius norms. However, under partition misspecification (\( S_k \in S^{(i)} \)), the eigenbasis approximation is not optimal because the true covariance under \( S^{(i)} \) is nonstationary. To be an optimal rank- \( m \) approximation, we must be able to construct the eigenbasis from the true nonstationary covariance function. On the other hand, thin plate spline is the least squares solution for an arbitrary smooth function (Wahba 1990). The thin plate splines \( \Phi_{kj}(s) = \| \mathbf{s} - \mathbf{u}_{kj} \|^2 \log(\| \mathbf{s} - \mathbf{u}_{kj} \|) \) used in our study are derived from a smoothness penalty function involving two-dimensional space. Therefore, the basis expansion \( \mathbf{\Phi}_k \mathbf{\delta}_k \) can accurately approximate the arbitrary true process \( \mathbf{W}_k \) regardless of partition specification. However, the resulting covariance function from SMB-SGLMMs, \( \sigma^2 \mathbf{\Phi}_k \mathbf{\Phi}_k^\top \) may be difficult to interpret, as opposed to the locally stationary models that provides an interpretable set of parameters (e.g., range, partial sill) for each partition.

### 3.3. Implementation Details

We need to specify (1) \( K \) number of partitions, (2) location of knots in each partition, and (3) a weighting radius \( \gamma \) for smoothing the local processes. In practice, we can set \( K \leq C \) (number of available cores) for parallel computation. Our method is heavily parallelizable, so computational walltimes tend to decrease with larger \( K \). For massive datasets, the lasso step can be computationally expensive for partitions \( S_k \) with large \( N_k \). In this case, we recommend subsampling the data using…
the following threshold. If $N_k > 0.02N$, we use 0.02N samples for basis selection for a partition $S_k$; otherwise, we use all $N_k$.

However, selecting a very large $K$ may result in unreliable local estimates due to a small number of observations $N_k$ within each partition. In our simulation study, we compare the performance of our approach with varying $K$. Then, we select the $K$ that minimizes the out-of-sample root cross-validated mean squared prediction error (rCVMSPE). Based on simulation results, the SMB-SGLMM is robust to the choice of $K$.

To avoid overfitting, we use lasso (Tibshirani 1996) to select the appropriate number and location of the knots. Initially, we set $m$ candidate knots $\{u_i\}_{i=1}^m$ uniformly over each partition $S_k$ (e.g., $m \approx 1000$). Then we fit a penalized glm with lasso using response $Z_k$ and covariates $[X_k, \Phi_k]$, where $\Phi_k$ is an $N_k$ by $m$ matrix. We impose an $l_1$ penalty to only the basis coefficients $\delta_k$, not the fixed effects $\beta_k$. We use the glmnet package (Friedman, Hastie, and Tibshirani 2010) in R for lasso regression. For basis selection, we choose the basis functions corresponding to the nonzero basis coefficients. Since we run lasso regression independently for each partition, this step is embarrassingly parallel.

From a pre-specified set of values (e.g., $\gamma = 0.01, 0.025, 0.05, 0.1$), we choose the $\gamma$ that yields the lowest rCVMSPE. Note that we choose $\gamma$ upon completion of Steps 1–3, the computationally demanding parts of SMB-SGLMM. Since the calculations in (6) are inexpensive, there are very little additional costs associated with Step 4.

### 3.4. Computational Complexity

Here, we examine the computational complexity of SMB-SGLMM. The three computationally demanding components are (a) basis selection (lasso), (b) MCMC for fitting the local processes, and (c) obtaining the global process. Here, parallelized computing is integral to the scalability of SMB-SGLMM. We provide the following discussion on computational costs and parallelization for each step. Table 1 summarizes complexity of SMB-SGLMM.

1. **Basis selection**: In each partition, our methods select the $m_k$ knots from $m$ candidates using a regularization method (lasso). Based on results in Friedman, Hastie, and Tibshirani (2010) the cost of the coordinate descent-based lasso is $O(N_k m)$, where $N_k$ is the number of observations in a partition $S_k$. We can select the basis functions for each partition in parallel across $K$ processors.

2. **MCMC for local processes**: The computational cost is dominated by matrix-vector multiplications $\Phi_k \delta_k$, where $\Phi_k$ is the $N_k$ by $m_k$ basis function matrix from the previous lasso step. The costs for this step is $O(N_k m_k)$. We can fit the local processes in parallel across $K$ processors.

### 3. Global process: We obtain the global process using weighted averages in (4). This step requires $O(N^2)$ complexity to calculate a distance matrix because the weights $c_k(s)$ in (4) are based on the distances between observations. Computing $c_k(s)$ requires a one-time computation of the distance matrix for all $N$ locations, which can be readily parallelized across $C$ available processors. We propose a novel way to "stream" the distances (supplementary materials) so that we can compute the weights $c_k(s)$ without actually storing the final distance matrix (e.g., 8TB for $N = 1$ million).

Considering that the complexity of the stationary SGLMM is $O(N^3)$, SMB-SGLMM is fast and provides accurate predictions for nonstationary processes.

### 4. Simulated Data Examples

We implement SMB-SGLMMs in massive nonstationary binary and ordered categorical datasets. We also provide a count data example in the supplementary materials. We implement our approach in nimble (de Valpine et al. 2017), a programming language for constructing Bayesian hierarchical models. Parallel computation is implemented through the parallel1 package in R. The computation times are based on a single 2.2 GHz Intel Xeon E5-2650v4 processor. All the code was run on the George Mason University Office for Research Computing (ORC) high-performance computing infrastructure (HOPPER).

**Simulation design**

Data is generated at 125,000 locations on the spatial domain $S \in \mathbb{R}^2$. We fit the spatial models using $N = 100,000$ observations and reserve the remaining $N_{cv} = 25,000$ observations for validation. We denote the model-fitting observations as $Z = \{Z(s_i) : s_i \in s\}$ where $s = \{s_1, \ldots, s_N\}$. Observations are generated using the SGLMM framework described in (1) with $\beta = (1, 1)$. The nonstationary spatial random effects $W = \{W(s_i) : s_i \in s\}$ are generated through convolving spatially varying kernel functions (Higdon 1998; Paciorek and Schervish 2006). For some $s \in s$ and reference locations $u_i \in D$, we have $W(s) = \sum_{j=1}^{I} K_i(u_j) V(u_i)$, where $K_i(u_j)$ is a spatially varying Gaussian kernel centered at reference location $u_j$ and $V(u_j)$ is a realization of Gaussian white noise. Additional details are provided in the Supplement. The binary dataset is generated using a logit link function, and the ordered categorical dataset is generated from a cumulative logit model.

We model the localized processes using the hierarchical framework in (3). To complete the hierarchical model, we set priors $\beta \sim N(0, 100I)$ and $\sigma^2 \sim IG(0.5, 2000)$. We implement SMB-SGLMM for different combinations of $K$ (the number of partitions) and $\gamma$ (weighting radius). We examine five partition groups $K = \{4, 9, 16, 25, 36\}$ and five weighting radii $\gamma = \{0.05, 0.1, 0.25, 0.5, 1\}$. For each case, we perform basis selection via lasso. We generate 100,000 samples from the posterior using a block random-walk Metropolis-Hastings algorithm using the adaptation routine from Shaby and Wells (2010). We examine predictive ability and computational cost. These include the root mean squared prediction error, misclassification rate, multi-class AUC, and the walltime required to run 100,000 iterations.

### Table 1

| Operations                  | Complexity                      |
|-----------------------------|---------------------------------|
| Basis selection             | $O(\sum_{k=1}^{K} N_k m_k / K)$ |
| MCMC                        | $O(\sum_{k=1}^{K} N_k m_k / K)$ |
| Weighted average            | $O(N^2 / C)$                    |
of the MCMC algorithm. We also present the posterior predictive mean surfaces.

The performance of SMB-SGLMM is robust in both nonstationary and stationary cases because the thin plate splines are derived from the ordinary least squares solution for an arbitrary process under the smoothness assumption. Similar to the nonstationary cases, we observe that SMB-SGLMM provides accurate predictions within a few minutes. We provide additional details and expanded discussion in the supplementary materials. To illustrate the performance of our approach, we compare SMB-SGLMM to two other competing methods: (a) the integrated nested Laplace approximation (INLA) (Rue, Martino, and Chopin 2009; Lindgren, Rue, and Lindström 2011); and (b) multi-resolution approach using nested bi-square basis functions (see Sengupta and Cressie 2013; Sengupta et al. 2016; Shi and Kang 2017). These two methods are computationally efficient and convenient to implement using existing software. We provide additional details of competing methods in the supplementary materials.

4.1. Binary Data

In Table 2, we present prediction results for the binary simulated dataset. For this example, we observe that increasing the number of partitions \( K \) and reducing the neighbor radius \( \gamma \) results in more accurate predictions and lower computational costs. Figure 2 includes the posterior predictive probability surface for the implementation yielding the lowest misclassification error \((K = 25, \gamma = 0.05)\). For the case where \( K = 25 \), the median number of basis functions per partition is 5 with a range of 0 to 53. In addition, computational walltimes decrease considerably as we increase the number of partitions. This is not surprising since we fit these models in parallel and the sample sizes for each partition tend to decrease as the number of total partitions increases. The localized parameter estimates of \( \beta \) are centered around the true parameter values \( \beta = (1, 1) \) (see supplementary materials).

We observe similar prediction performance across the different approaches (Table 2). Compared to the count data example (Supplement), model-fitting walltimes are shorter in the binary case. These shorter walltimes are a result of using fewer spatial basis functions (thin plate splines) per partition in the binary example. SMB-SGLMM \((K = 25, \gamma = 0.05)\) is faster than the multiresolution approach, though slower than INLA. However, the basis function approaches generate samples from the joint posterior distribution, while INLA provides approximations of the marginal posteriors only. Furthermore, basis function approaches can be readily extended to application-specific hierarchical spatial models, which may not be included in INLA’s current suite of models.

4.2. Ordered Categorical Data

Table 3 provides prediction results for the simulated ordered categorical dataset. We observe that increasing the number of partitions \( K \) decreases computational walltimes considerably for the reasons presented in the previous section. The performance of SMB-SGLMM is robust across different neighbor radius \( \gamma \) values. Figure 3 illustrates the posterior predictive probability surface for the implementation with the multi-class area under the receiver operating curve (ROC) \((K = 36)\). For \( K = 36 \), the median number of basis functions per partition is 32 with a range of 19–89.

![Probability (Truth)](image1)

![Probability (Predicted)](image2)
Table 3. Multi-class area under the ROC curve (AUC) and total walltime (minutes) for the ordinal data simulated example.

| SMB-SGLMM | Weighting radius (γ) | Walltime (minutes) |
|-----------|----------------------|-------------------|
| Partitions | 0.05  | 0.1 | 0.25 | 0.5 | 1 |
| 4 | 0.910 | 0.910 | 0.910 | 0.910 | 118.349 |
| 9 | 0.967 | 0.967 | 0.967 | 0.967 | 78.325 |
| 16 | 0.978 | 0.978 | 0.978 | 0.978 | 44.513 |
| 25 | 0.982 | 0.982 | 0.982 | 0.982 | 44.329 |
| 36 | 0.982 | 0.982 | 0.982 | 0.982 | 30.208 |
| Bisquare | 0.976 | 169.481 |

NOTE: For SMB-SGLMM, the rows denote the five partition classes and columns correspond to the chosen weighting radius (γ). Walltimes denote the combined walltimes for lasso, MCMC, and weighting. The final row includes the AUC and walltime for the multi-resolution approach.

Since INLA is not available for the ordered categorical dataset, we only compare our method to the multiresolution approach. As in the previous example, we observe similar prediction performance across the different approaches (Table 3). We observe longer model-fitting walltimes than in the binary data example because we select more thin plate spline basis functions per partition in the ordered categorical data example. We observe that SMB-SGLMM (K = 36) is much faster than the multiresolution approach due to parallelization. As shown in this example, both basis function approaches can be embedded into customized hierarchical spatial models for ordered categorical data.

5. Applications

In this section, we apply our method to two remote sensing datasets collected by the NASA. For both large non-Gaussian nonstationary datasets, SMB-SGLMM provides accurate predictions within a reasonable timeframe. We also provide an additional water vapor data example in the supplementary materials.

5.1. Cloud Mask Data

NASA launched the Terra Satellite in 1999 as part of the Earth Observing System. As in past studies (Sengupta and Cressie 2013; Bradley, Holan, and Wikle 2020), we model the cloud mask data (Platnick et al. 2003) captured by MODIS instrument onboard the Terra satellite. The response is a binary incidence of cloud mask at a 1 km × 1 km spatial resolution. In this study, we selected N = 2,473,758 observations to fit our model and reserved Ncv = 274,862 for validation. As in Sengupta and Cressie (2013); Bradley, Holan, and Wikle (2020), we include the vector 1 and a vector of latitudes as the covariates and use a logit link function.

For the SMB-SGLMM approach, we vary the number of partitions K ∈ {16, 25, 36, 49, 64, 81} and weighting radius γ ∈ {0.01, 0.025, 0.05, 0.1} for a total of 24 cases. For each partition, we begin with m = 1000 knots and perform basis selection using lasso Tibshirani (1996). On average, basis selection results in roughly 21.4 basis functions per partition. For each partition, we fit a local model (3) by running the MCMC algorithm for 100,000 iterations.

Figure 4 indicates that validation binary observations and predicted probability surface exhibit similar spatial patterns. We also provide the misclassification rate for each case in Table 4. The performance of SMB-SGLMMs is robust across different choices of K and γ. Results suggest that a larger number of partitions (K = 81) and a smaller weighting radius (γ = 0.01) yields the most accurate predictions. The combined walltimes roughly decrease when using more partitions. We observe that SMB-SGLMM (K = 81 and γ = 0.01) provides the most accurate predictions compared to the other competitors. As in the simulated data examples, SMB-SGLMM is faster than the multiresolution approach, and INLA has shorter walltimes than the other two approaches.

For SMB-SGLMM with K = 81, γ = 0.01, we observe the false positive rate as 0.09 and the false negative rate as 0.21. It is worth pointing out that the latent conjugate distribution (Bradley, Holan, and Wikle LCM, 2020) takes about 12 hr with the false positive rate 0.22 and the false negative rate 0.28 for the MODIS cloud mask data example based on Bradley, Holan, and Wikle (2020). Bradley, Holan, and Wikle (2020) also report that the support vector machine (Hastie, Tibshirani, and Friedman SVM, 2009) takes about 72 hr with the false positive rate 0.11 and the false negative rate 0.53. A direct comparison between
SMB-SGLMM and these reported results may not be fair due to differences in code optimization and the available computing resources; however, this implies that SMB-SGLMM is scalable and provide comparable prediction performances compared to the state-of-the-art competitors.

### 5.2. Standardized Precipitation Index Data

We apply our method to the Standardized Precipitation Index (SPI) calculated from 20 years of precipitation records. A monthly averaged precipitation rate for each location with 0.1-degree resolution is collected by NASA's two satellites, the Tropical Rainfall Measuring Mission (TRMM) and Global Precipitation Measurement (GPM) mission. The observations from these two satellites are then calibrated by NASA's Integrated Multi-satellite Retrievals for GPM (IMERG) algorithm that provides the multi-satellite precipitation product (Huffman et al. 2019). The SPI is a drought index used for estimating dry or wet conditions solely based on precipitation. Each location is rated Wet (1 or more), Mildly Wet (0 to 0.99), Mildly Drought (−0.99 to 0), Drought (−1 or less). We use the data with the latitude range (−180, 0) and the longitude range (−60, 60), resulting in 2,161,800 locations. The dataset can be downloaded from [https://disc.gsfc.nasa.gov/](https://disc.gsfc.nasa.gov/). For our implementation, we include the vector of latitudes and longitudes as the covariates.

We use $N = 1,836,000$ observations to train the model and reserve $N_{cv} = 324,000$ observations for validation. For the covariates, we use the locations’ latitudes and longitudes. We investigate the performance of our method across different $K \in \{16, 25, 36, 49, 64, 81\}$ and $\gamma \in \{0.05, 0.1, 0.25, 0.5, 1\}$. We implement lasso (Tibshirani 1996) to select basis functions, which range from 4115 to 20,628 depending on $K$. On average, we obtain roughly 26.3 basis functions per partition. For each partition, we fit a local model (3) by running the MCMC algorithm for 100,000 iterations.

We compare the misclassification rate for each case (Table 5). As in the previous examples, SMB-SGLMM shows robust performance across different $K$ and $\gamma$ values. As we increase $K$,
Figure 5. True (left) and predicted SPI drought index groups using SMB-SGLMM (center) and multi-resolution basis functions (right). We use $K = 81$ and $\gamma = 0.05$ for the SMB-SGLMM implementation. The multi-class AUC is 0.846 for SMB-SGLMM and 0.739 for the multi-resolution approach.

the combined walltimes tend to decrease; we can fit a “locally nonstationary” process to smaller datasets. For this example, we observe that $K = 81$, $\gamma = 0.05$ provides the highest multi-class area under the receiver operating characteristic (ROC) curve. Figure 5 indicates that the predicted mean surface shows similar patterns as the true observations. SMB-SGLMM exhibits a dramatic decrease in computational costs over the multiresolution approach with a higher multi-class AUC value. As we pointed out in Section 4, both basis function approaches are available to a wide variety of hierarchical models, while INLA can be limited.

6. Discussion

We propose a scalable algorithm for modeling massive nonstationary non-Gaussian datasets. Our method divides the spatial domain into disjoint partitions using a spatial clustering algorithm (Heaton, Christensen, and Terres 2017). For each partition, we fit a localized model using thin plate spline basis expansions that can capture the nonstationary behavior. We provide a basis selection process via a regularization approach. This framework is efficient due to parallel computing and using basis representations of complex spatial processes. Our study shows that the proposed method provides accurate estimations and predictions within a reasonable time. In the simulated and the real data examples, the SMB-SGLMM approach provides comparable (or better) predictive performance than the competing methods. Compared to other approaches, SMB-SGLMM provides the most accurate predictions for massive cases. Our method is attractive due to its flexibility in fitting a wide array of hierarchical spatial models and its scalability to massive datasets.

Another important contribution of this article is developing a locally nonstationary Bayesian framework for non-Gaussian spatial data. As we pointed out, SMB-SGLMM can be more robust against partition misspecification than many locally stationary models (see Paciorek and Schervish 2006; Anderes and Stein 2011). Under the appropriate smoothness assumption, thin plate splines can accurately recover the arbitrary true process. We note that quantifying and comparing approximation errors between local stationary and local nonstationary models would be problem-specific, depending on the smoothness of the true latent process, choice of basis functions, and other modeling considerations. Developing statistical tools for a direct comparison would be a natural extension of this study.

One of the main advantages of our method is its flexibility, where SMB-SGLMM is extendable to a wide variety of Bayesian hierarchical models with minor changes to the nimble code. Examples of such models include a spatial model for non-Gaussian data fusion (Shi and Kang 2017), complex hierarchical model for functional magnetic resonance imaging data (Musgrove, Hughes, and Eberly 2016), and ordered categorical (or multinomial) spatial regressions (Lee and Haran 2021). SMB-SGLMM can be readily extended to many hierarchical spatial models, as long as there exists a spatially-dependent process within the process model level. Therefore, practitioners can readily use SMB-SGLMM to fit the application-specific hierarchical spatial model of their choice.

The proposed framework can be extended to a wider range of spatial basis functions. In the literature, there exists a wide array of spatial basis functions such as bi-square (radial) basis functions using varying resolutions (Cressie and Johannesson 2008; Nychka et al. 2015; Katsfuss 2017), empirical orthogonal functions (Cressie 2015), predictive processes (Banerjee et al. 2008), Moran’s basis functions (Griffith 2003; Hughes and Haran 2013), wavelets (Nychka, Wikle, and Royle 2002; Shi and Cressie 2007), Fourier basis functions (Royle and Wikle 2005) and Gaussian kernels (Higdon 1998). A closer examination of adopting Bayesian regularization methods (see O’Hara and Sillanpää 2009 for a detailed review) for selecting basis functions is also an interesting future research avenue.

Developing scalable methods for modeling nonstationary non-Gaussian spatio-temporal data is challenging. The partition-based basis function representation can be integrated into existing hierarchical spatio-temporal models. For example, we can approximate the nonstationary processes using a...
tensor product of spatial and temporal basis functions. Agent-based models (Hooten and Wikle 2010; McDermott, Wikle, and Millsapagh 2017) are also useful for describing complex non-Gaussian spatio-temporal processes. They represent non-stationary anisotropic processes by using covariate information into the neighbor transition probabilities. These approaches are viable alternatives to our modeling framework.

Supplementary Materials

Supplement contains source code, details for the spatial clustering algorithm, data generation in the simulated examples, parallelized schemes, and additional simulations for count data. It also includes partition-varying $\beta$ estimates, details for the comparative analysis, a variant of SMB-SGLMM with geometric median weights, and a water vapor data example.

Acknowledgments

The authors are grateful to Matthew Heaton, Murali Haran, John Hughes, and Whitney Huang for providing useful sample code and advice. The authors are thank the anonymous reviewers for their careful review and valuable comments.

Disclosure Statement

The authors report there are no competing interests to declare.

Funding

Jaewoo Park was supported by the Yonsei University Research Fund 2020-22-0501 and the National Research Foundation of Korea (NRF-2020R1C1A1010038611).

References

Anderes, E. B., and Stein, M. L. (2011), “Local Likelihood Estimation for Nonstationary Random Fields,” Journal of Multivariate Analysis, 102, 506–520. [109,114]
Bakka, H., Rue, H., Fuglstad, G.-A., Riebler, A., Bolin, D., Illian, J., Krainsi, E., Simpson, D., and Lindgren, F. (2018), “Spatial Modeling with R-INLA: A Review,” Wiley Interdisciplinary Reviews: Computational Statistics, 10, e1443. [107]
Banerjee, A., Dunson, D. B., and Tokdar, S. T. (2013), “Efficient Gaussian Process Regression for Large Datasets,” Biometrika, 100, 75–89. [108,109]
Banerjee, A., and Gelfand, A. E. (2006), “Bayesian Wombling: Curvilinear Gradient Assessment under Spatial Process Models,” Journal of the American Statistical Association, 101, 1487–1501. [107]
Banerjee, S., Gelfand, A. E., Finley, A. O., and Sang, H. (2008), “Gaussian Predictive Process Models for Large Spatial Data Sets,” Journal of the Royal Statistical Society, Series B, 70, 825–848. [106,108,114]
Bradley, J. R., Cressie, N., and Shi, T. (2016), “A Comparison of Spatial Predictors When Datasets Could Be Very Large,” Statistics Surveys, 10, 100–131. [106]
Breiding, J. R., Holan, S. H., and Wikle, C. K. (2020), “Bayesian Hierarchical Models with Conjugate Full-Conditional Distributions for Dependent Data from the Natural Exponential Family,” Journal of the American Statistical Association, 115, 2037–2052. [107,112]
Bradley, J. R., Holan, S. H., Wikle, C. K. (2018), “Computationally Efficient Multivariate Spatio-Temporal Models for High-Dimensional Count-Valued Data” (with Discussion), Bayesian Analysis, 13, 253–310. [107]
Campbell, T., and Broderick, T. (2019), “Automated Scalable Bayesian Inference via Hilbert Coresets,” The Journal of Machine Learning Research, 20, 551–588. [108]
Cressie, N. (2015), Statistics for Spatial Data, New York: Wiley. [108,114]
Cressie, N., and Johannesson, G. (2008), “Fixed Rank Kriging for Very Large Spatial Data Sets,” Journal of the Royal Statistical Society, Series B, 70, 209–226. [114]
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. [106,110]
Diggel, P. I., Tawn, J., and Moyeed, R. (1998), “Model-based Geostatistics,” Journal of the Royal Statistical Society, Series C, 47, 299–350. [105,106]
Fouedjio, F., Desassis, N., and Rivoirard, J. (2016), “A Generalized Convolu- tion Model and Estimation for Non-stationary Random Functions,” Spatial Statistics, 16, 35–52. [109]
Friedman, J., Hastie, T., and Tibshirani, R. (2010), “Regularization Paths for Generalized Linear Models via Coordinate Descent,” Journal of Statistical Software, 33, 1–22. [110]
Fuentes, M. (2001), “A High Frequency Kriging Approach for Non-stationary Environmental Processes,” Environmetrics: The Official Journal of the International Environmetrics Society, 12, 469–483. [105,106,108,109]
——(2002), “Interpolation of Nonstationary Air Pollution Processes: A Spatial Spectral Approach,” Statistical Modelling, 2, 281–298. [105]
Gelfand, A. E., Kim, H.-J., Sirmans, C., and Banerjee, S. (2003), “Spatial Modeling with Spatially Varying Coefficient Processes,” Journal of the American Statistical Association, 98, 387–396. [109]
Griffith, D. A. (2003), “Spatial Filtering” in Spatial Autocorrelation and Spatial Filtering, pp. 91–130, Berlin: Springer. [114]
Guan, Y., and Haran, M. (2018), “A Computationally Efficient Projection-based Approach for Spatial Generalized Linear Mixed Models,” Journal of Computational and Graphical Statistics, 27, 701–714. [106,108]
Guhaniyogi, R., and Banerjee, S. (2018), “Meta-kriging: Scalable Bayesian Modeling and Inference for Massive Spatial Datasets,” Technometrics, 60, 430–444. [105]
Haran, M., Hodges, J. S., and Carlin, B. P. (2003), “Accelerating Computation in Markov Random Field Models for Spatial Data via Structured MCMC,” Journal of Computational and Graphical Statistics, 12, 249–264. [108]
Hastie, T., Tibshirani, R., and Friedman, J. (2009), The elements of Statistical Learning: Data Mining, Inference and Prediction, New York: Springer. [112]
Heaton, M. J., Christensen, W. F. E., and Terres, M. A. (2017), “Nonstationary Gaussian Process Models using Spatial Hierarchical Clustering from Finite Differences,” Technometrics, 59, 93–101. [105,106,107,114]
Hefley, T. J., Broms, K. M., Brost, B. M., Buderman, F. E., Kay, S. L., Scharf, H. R., Tipton, J. R., Williams, P. J., and Hooten, M. B. (2017), “The Basis Function Approach for Modeling Autocorrelation in Ecological Data,” Ecology, 98, 632–646. [105]
Higdon, D. (1998), “A Process-Convolution Approach to Modelling Tem- peratures in the North Atlantic Ocean,” Environmental and Ecological Statistics, 5, 173–190. [105,106,108,110,114]
Higdon, D., Gattiker, J., Williams, B., and Rightley, M. (2008), “Computer Model Calibration using High-Dimensional Output,” Journal of the American Statistical Association, 103, 570–583. [108]
Holland, D. M., Saltzman, N., Cox, L. H., and Nychka, D. (1999), “Spatial Prediction of Sulfur Dioxide in the Eastern United States,” in geoENV II–Geostatistics for Environmental Applications, eds. J. Gómez-Hernández, A. Soares, and R. Froidevaux, pp. 65–76, Dordrecht: Springer. [106,108]
Hooten, M. B., and Wikle, C. K. (2010), “Statistical Agent-based Models for Discrete Spatio-Temporal Systems,” Journal of the American Statistical Association, 105, 236–248. [115]
Huffman, G., Stocker, E., Bolvin, D., Nelkin, E., and Tan, J. (2019), GPM MERG Final Precipitation L3 1 month 0.1 degree x 0.1 degree v06, Greenbelt, MD, Goddard Earth Sciences Data and Information Services Center (GES DISC). [113]
Hughes, J., and Haran, M. (2013), “Dimension Reduction and Allievation of Confounding for Spatial Generalized Linear Mixed Models,” Journal of the Royal Statistical Society, Series B, 75, 139–159. [108,114]
Ingebritsen, R., Lindgren, F., and Steinsland, I. (2014), “Spatial Models with Explanatory Variables in the Dependence Structure,” Spatial Statistics, 8, 20–38. [107]
