Taylor & Francis
Taylor & Francis Group

Journal of Computational and Graphical Statistics

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/ucgs20

Partition-Based Nonstationary Covariance Estimation Using the Stochastic Score Approximation

Amanda Muyskens, Joseph Guinness & Montserrat Fuentes

To cite this article: Amanda Muyskens, Joseph Guinness & Montserrat Fuentes (2022) Partition-Based Nonstationary Covariance Estimation Using the Stochastic Score Approximation, Journal of Computational and Graphical Statistics, 31:4, 1025-1036, DOI: 10.1080/10618600.2022.2044830

To link to this article: https://doi.org/10.1080/10618600.2022.2044830

JOURNAL OF COMPUTATIONAL AND GRAPHICAL STATISTICS
2022, VOL. 31, NO. 4, 1025–1036
https://doi.org/10.1080/10618600.2022.2044830

Partition-Based Nonstationary Covariance Estimation Using the Stochastic Score Approximation

Amanda Muyskens[a,*], Joseph Guinness[b,*], and Montserrat Fuentes[c,*,**]

[a]Applied Statistics Group, Lawrence Livermore National Lab, Livermore, CA; [b]Department of Statistics, Cornell University, Ithaca, NY; [c]Office of the President, University of Iowa, Iowa City, IA

ABSTRACT

We introduce computational methods that allow for effective estimation of a flexible nonstationary spatial model when the field size is too large to compute the multivariate normal likelihood directly. In this method, the field is defined as a weighted spatially varying linear combination of a globally stationary process and locally stationary processes. Often in such a model, the difficulty in its practical use is in the definition of the boundaries for the local processes, and therefore, we describe one such selection procedure that generally captures complex nonstationary relationships. We generalize the use of a stochastic approximation to the score equations in this nonstationary case and provide tools for evaluating the approximate score in $O(n \log n)$ operations and $O(n)$ storage for data on a subset of a grid. We perform various simulations to explore the effectiveness and speed of the proposed methods and conclude by predicting average daily temperature. Supplementary materials for this article are available online.

ARTICLE HISTORY
Received July 2018
Revised December 2021

KEYWORDS
Circulant embedding; Computational efficiency; Gridded data; Prediction; Spectral density; Temperature

1. Introduction

Gaussian process models are commonly used to account for correlation in data collected over space, time, or in computer experiments. Addressing modern problems in environmental science often requires data collected on large domains, such as over the United States or across the world. Therefore, second order nonstationary models are often needed to capture the complexity of the spatial relationships of environmental variables, as these large spatial domains often have varied underlying climate conditions, spatially varying topography, and other environmental or anthropogenic factors. However, when the number of observations (n) in these environmental datasets is large ($n > 10{,}000$), formation of the covariance matrix is often prohibited due to memory or computing time limits. Therefore, defining nonstationary covariance functions and developing corresponding computational methods are an integral part of environmental research and must be done in a matrix-free way.

Our approach for modeling nonstationary processes is similar to that in Fuentes (2001, 2002), but our new estimation method allows for more flexible applications. Let $\{s_1, s_2, \ldots, s_n\}$ be the spatial locations where data are observed. Fuentes models the univariate nonstationary process $Y(s_1)$ at spatial location s_1 as a weighted sum of stationary processes:

$$Y(s_1) = \sum_{h=0}^{q} \omega_h(s_1) Z_h(s_1), \tag{1}$$

where $Z_h \sim GP(\mu_h, C_h)$ are assumed independent over $h = 0, 1, 2, \ldots, q$, and C_h are stationary covariance functions with parameter vectors θ_h. The ω_h are assumed to be nonrandom, unknown, and nonnegative spatially varying weights. The covariance of observations $Y(s_1)$ and $Y(s_2)$ at spatial locations s_1 and s_2 is

$$K_\theta(s_1, s_2) = \sum_{h=0}^{q} \omega_h(s_1)\omega_h(s_2) C_h(s_1, s_2). \tag{2}$$

This is a valid positive definite covariance since it is the linear combination of covariances C_h.

As in Fuentes (2001), throughout most of our article, we assume that $\omega_1, \ldots, \omega_q$ are indicator functions, defined as

$$\omega_h(s_1) = I\{s_1 \in D_h\}, \tag{3}$$

where $D = \{D_1, \ldots, D_q\}$ is a partition of the spatial domain. However, we provide computational methods that allow us to extend the Fuentes (2001) approach in several ways. First, we set $\omega_0(s_i) = 1$ for all $i = 1, 2, \ldots, n$, which introduces dependence across blocks of the partition (Fuentes, Chaudhuri, and Holland 2007). Further, this extends the nonstationary model to include a stationary process as a special case; namely, when the variances of $Z_1, \ldots, Z_q$ are zero. Second, whereas Fuentes (2001) used rectangular blocks D_h to facilitate taking local discrete Fourier transforms within each block, we allow the blocks to take on more arbitrary shapes.

Spatial partitions are a popular mechanism for specifying nonstationary models. For example, Risser et al. (2016) use a partition model with blocks defined via covariate information.

CONTACT Amanda Muyskens muyskens1@llnl.gov Applied Statistics Group, Lawrence Livermore National Lab, Livermore, CA.
*Present affiliation: President of St. Edward's University in Austin, TX.
**Previous affiliation: Department of Statistics, North Carolina State University, Raleigh, NC.
Supplementary materials for this article are available online. Please go to www.tandfonline.com/r/JCGS.

Park and Apley (2018) partition the domain with a method based on the principal components directions, using equally sized blocks. Heaton, Christensen, and Terres (2017) use the dissimilarity of finite differencing to define partition blocks. In Gramacy and Lee (2008) and Konomi, Sang, and Mallick (2014) a treed partitioning structure is used. All of these methods assume either that the process is independent across blocks or use local likelihoods to estimate parameters within each block. This requires the number of points within each block to be small enough to allow for formation of the within-block covariance matrices or the shape of the blocks is restricted so that further approximations can be applied. Our methods allow for dependence across blocks, irregularly shaped blocks, and more diverse block sizes. This increased flexibility is made possible through a generalization of Stein, Chen, and Anitescu (2013) stochastic score approximation to nonstationary models. The methods make use of fast Fourier transforms (FFTs) and circulant embedding, and thus, are technically applicable only to gridded data, though we do allow for missing values and irregularly shaped domains. The temperature data we analyze do not fall on a regular grid, but we demonstrate that they can be analyzed using our methods after mapping the data to a high resolution grid.

There are numerous alternative nonstationary models, and no model will perform uniformly better than others for all applications. In kernel convolution (Higdon 1998), spatially-varying kernel functions are convolved with a stationary, often white noise, process. Fixed Rank Kriging (Cressie and Johannesson 2008) is a similar approach that defines a process in terms of a small number of basis functions to limit computational cost. Lattice Kriging (Nychka et al. 2015) is also a basis function approach but differs in that a large number of basis functions are used. It employs compactly supported basis functions and a Markov random field assumption on the basis coefficients in order to reduce computational cost. Deformation models require the formation of full covariance matrices and are therefore, computationally inefficient for large datasets (Sampson and Guttorp 1992). Moving window models (Haas 1995; Datta et al. 2016) are efficient because their computation is entirely parallelizable, but there is no guarantee that the resulting global model covariance is positive definite. Prediction using some of these models as well as many other computationally efficient stationary and nonstationary methods are compared in Heaton et al. (2018).

The main contributions of our article are the following. We introduce a method for estimating the partition of a nonstationary model by generating an ensemble of candidate partitions and choosing one from the ensemble via an information criterion (Section 2). We adapt the stochastic score method (Stein, Chen, and Anitescu 2013) to our nonstationary case, which allows us to handle large datasets and arbitrary partitions (Section 3). Computational details for prediction are discussed in Section 4. Simulation results are presented in Section 5. In our analysis of daily temperature data in Section 6, we consider an additional extension to the model, relaxing the indicator weight assumption to allow for smoother transitions across blocks.

2. Partition Estimation Method

The first step in our estimation procedure is to use the data to estimate a partition of the domain $D = \{D_1, D_2, \ldots, D_q\}$ defining spatially contiguous subregions of local stationarity. Since enumerating all possible partitions is intractable, our strategy is to first generate a diverse ensemble of candidate partitions and use an information criterion to choose the best partition from the ensemble. In Fuentes (2001) the partition is chosen via AIC or BIC from an ensemble of candidates of only equally sized blocks. Alternatively, Gramacy and Lee (2008) and Konomi, Sang, and Mallick (2014) use a treed partition with nested breaks parallel to the axes. Guinness and Fuentes (2015) use the Ising model to uncover stationary partition blocks of the domain using spectral techniques. We propose a procedure that can return more flexible partition shapes and is based on repeated likelihood ratio testing. A comparison of the flexibility of our partition candidates to some existing methods is in Appendix D, supplementary materials.

To generate a member of the partition ensemble, we begin by partitioning the domain into a fine base partition $B = \{B_1, B_2, \ldots, B_p\}$ with p equally sized square blocks, where p is large compared to the expected number of partition blocks. See Figure 1 for an example. Then we iteratively build our candidate partition by joining pairs of neighboring blocks according to tests of local stationarity across the blocks. The iterative algorithm is stopped after looping over all pairs of neighboring blocks. This process is repeated many times with different orderings of the pairs of blocks and different stringencies on the tests, resulting in a diverse ensemble of partitions from which a best partition can be selected via an information criterion. The

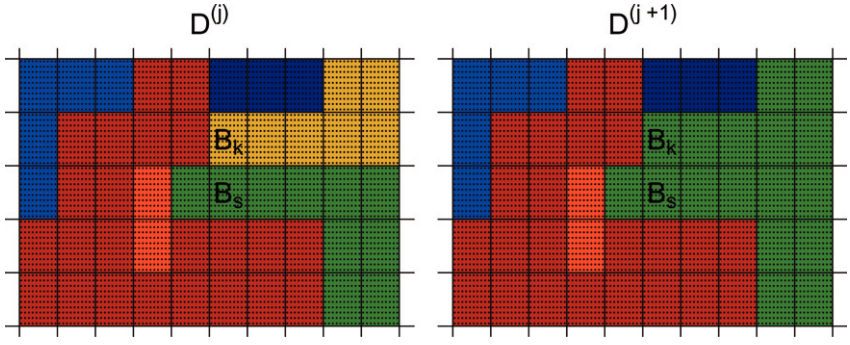

Figure 1. Example of block joining step of B_k and B_s at the $(j+1)$th step of the partition estimation algorithm assuming we cannot reject H_0. Shade represents membership of the block to a partition $D_i^{(j)}$ and $D_i^{(j+1)}$. $D_\ell^{(j)}$ is the green region and $D_m^{(j)}$ is the yellow region. The black lines define the base partition B.

choice of base partition is ad hoc, but we show in Section 6 that since these blocks are small in comparison to the data's large spatial domain, predictions using two different resolution base partitions are similar.

More formally, the iterative method is initialized with $D^{(0)} = B$ so that $D_i^{(0)} = B_i$ for $i = 1, 2, \ldots, p$. Let $D^{(j)}$ be the state of the partition at iteration j, so that $D_m^{(j)}$ (the mth block of $D^{(j)}$) consists of a union of B_is. Define the likelihood

$$L_m^{(j)}(\theta) = \prod_{B_i \subset D_m^{(j)}} L(\theta | B_i)$$

to be the product of the individual likelihoods of data within each block B_i within $D_m^{(j)}$, and let $\theta_m^{(j)}$ be the maximizer of this likelihood. Given a base partition B, define the set N_B as the set of all pairs of neighbor blocks $N_B = \{\{B_s, B_k\} | B_s \text{ neighbors } B_k\}$ with n_B elements ordered randomly. Suppose that (B_s, B_k) is the $(j+1)$th pair of neighboring blocks in N_B. If B_s and B_k are already members of the same block of $D^{(j)}$, we set $D^{(j+1)} = D^{(j)}$ and move to the next iteration. If B_s and B_k are in different blocks of $D^{(j)}$, say $B_s \in D_\ell^{(j)}$ and $B_k \in D_m^{(j)}$, we test the hypotheses

$$H_0 : \theta_\ell = \theta_m \quad \text{versus} \quad H_1 : \theta_\ell \neq \theta_m,$$

using the likelihood ratio statistic

$$\frac{L_\ell^{(j)}(\theta^*) L_m^{(j)}(\theta^*)}{L_\ell^{(j)}(\theta_\ell^{(j)}) L_m^{(j)}(\theta_m^{(j)})}, \tag{4}$$

where θ^* is the maximizer of $L_\ell^{(j)}(\theta) L_m^{(j)}(\theta)$. If we reject the null hypothesis at level α, we keep B_s and B_k in separate blocks and set $D^{(j+1)} = D^{(j)}$; if we fail to reject, then we join blocks $D_\ell^{(j)}$ and $D_m^{(j)}$ in a block in $D^{(j+1)}$. For all other elements $i \neq \ell$ or m, set $D_i^{(j+1)} = D_i^{(j)}$. Figure 1 illustrates the joining of blocks. The iterative method is stopped after looping over all pairs in N_B, resulting in a candidate partition $D = D^{(n_B)}$ that has q partition blocks with parameter vectors $\theta_h = \theta_h^{(n_B)}$ for $h = 1, 2, \ldots, q$. Finally, define the likelihood statistic

$$L(D) = \prod_{h=1}^{q} L_h^{(n_B)}(\theta_h) \tag{5}$$

to be the product of individual likelihoods over all blocks of D.

This one candidate partition depends on the significance level chosen for the likelihood ratio tests and the ordering of the neighbor pairs in N_B. Thus, we suggest repeating this entire procedure with different randomization orders in N_B as many times as possible given computing constraints to obtain a set of nonnested candidate partitions. In order to produce a diverse set of candidate partitions, the significance level in the likelihood ratio testing should be varied in the interval $[\frac{0.05}{n_B}, 0.01]$. From this ensemble, we use the likelihood statistics to select the partition that minimizes the BIC using the likelihood in Equation (5). Simulations of the effectiveness of this approximate partition selection method can be seen in Section 5.2. If smoother predictions are desired across block definitions, smoothing may be applied to the edges of the partition blocks after this partition estimate algorithm is performed. Further details on one possible block smoothing procedure are described in Section 6.

3. Efficient Computation of the Stochastic Score

Given estimation of D from Section 2, nonsmooth block weights ω_h are fixed as defined in Equation (3). This section describes our computational methods for estimating the mean and covariance parameters of the independent Gaussian processes Z_h for $h = 0, 1, \ldots, q$. Typical maximum likelihood estimation involves an optimization algorithm such as Newton's method or gradient descent, where the score and likelihood are evaluated repeatedly. However, a large number of observations ($n >$ 10,000) often prevents the necessary formation of the covariance matrix. Therefore, in this section, we detail a matrix-free stochastic approximation to the score that is implemented in a parameter estimation algorithm similar to a gradient descent algorithm. Further details of the algorithm are in Appendix C, supplementary materials.

We extend the Stein, Chen, and Anitescu (2013) stochastic score approximation to nonstationary covariance estimation. Let $\theta^T = [\theta_0^T, \ldots, \theta_q^T]$ be the concatenated vector of all of the covariance parameters, with θ_h being the vector of parameters for C_h, and K_θ be the $n \times n$ covariance with entries as defined in Equation (2). Define n_p to be the length of this θ vector, which in the case the same covariance form is used for all partition blocks, is $(q+1)\times$ the number of parameters in C_h. Define $Y_0(s_1)$ as the de-trended observation at spatial location s_1 as

$$Y_0(s_1) = Y(s_1) - E(Y(s_1)) = Y(s_1) - \sum_{h=0}^{q} \omega_h(s_1) \mu_h(s_1). \tag{6}$$

Further, let $Y_0^T = [Y_0(s_1), Y_0(s_2), \ldots, Y_0(s_n)]$. Stein, Chen, and Anitescu (2013) approximate the rth value in the score of the multivariate normal log-likelihood by:

$$S_r(\theta | Y_0) \approx \widetilde{S}_r(\theta | Y_0) = \frac{1}{2} Y_0^T K_\theta^{-1} K_\theta^{(r)} K_\theta^{-1} Y_0 - \frac{1}{2N} \sum_{j=1}^{N} U_j^T K_\theta^{-1} K_\theta^{(r)} U_j, \tag{7}$$

where $K_\theta^{(r)}$ is the $n \times n$ partial derivative matrix of the covariance matrix with respect to the rth parameter in θ for $r = 1, 2, \ldots, n_p$. Define the random vector $U_j^T = [U_j(1), \ldots, U_j(n)]$ with $U_j(k) \sim$ Bernoulli$(1/2, -1, 1)$ and $U_j(k)$ independent of $U_j(\ell)$ for $k \neq \ell$. These assumptions imply that the expected value of the approximate score is zero, making $\widetilde{S}_r(\theta | Y_0)$ a set of unbiased estimating equations for every parameter θ. We further take U_j to be independent of U_k for $j \neq k$. Stein, Chen, and Anitescu (2013) considered allowing for dependence between U_j and U_k for $j \neq k$, but we did not find that dependent sampling improved our estimates significantly. The impact of the choice of N will be evaluated in simulation in Section 5. This stochastic formulation is convenient because formation of the covariance matrix is not necessary; only matrix-vector multiplications and solves are required. Stein, Chen, and Anitescu (2013) considered a stationary model for data on a regular grid. This allows for the use of circulant embedding techniques to evaluate Equation (7). In subsequent sections, we develop extensions of these methods to this nonstationary model for data observed on part of a grid.

Following the recommendation of Guinness and Fuentes (2017), we model the locally and globally stationary covariances (C_h) with the quasi-Matérn spectral density. Define the

covariance of the Gaussian processes through their spectral representation

$$C_h(s_1, s_2) = \int_{[0,2\pi]^2} e^{i(s_2-s_1)'\gamma} f_h(\gamma) d\gamma. \quad (8)$$

Define $\theta_h = [\sigma_h^2, \alpha_h, \nu_h, \tau_h]^T$. Then for scale σ_h^2, range α_h, smoothness ν_h, and nugget τ_h parameters, the quasi-Matérn spectral density for two-dimensional fields is

$$f_h(\gamma) = c_h(1-\tau_h)\sigma_h^2 \left\{1 + \alpha_h^2 \left[\sin^2\left(\frac{\gamma^1}{2}\right) + \sin^2\left(\frac{\gamma^2}{2}\right)\right]\right\}^{-1-\nu_h} + \tau_h, \quad (9)$$

where c_h is the normalizing constant and $\gamma = (\gamma^1, \gamma^2)$ are the two-dimensional Fourier frequencies defined on the region $[0, 2\pi]^2$.

3.1. Preconditioned Conjugate Gradient

The primary computational burden of computing the stochastic score itself is in the linear solves of the form $K_\theta x = y$, where $y = Y_0$ or U_j. Since the covariance matrix is symmetric and positive definite, we can apply the preconditioned conjugate gradient algorithm to approximate x within a threshold of accuracy (Hestenes and Stiefel 1952). Preconditioned conjugate gradient is an iterative solving method computationally dominated by matrix-vector multiplication, where instead of solving the linear system $K_\theta x = y$, we solve the equivalent equations $P_\theta K_\theta x = P_\theta y$ where P_θ is an $n \times n$ preconditioning matrix. In order for P_θ to be an effective preconditioner, the condition number of $P_\theta K_\theta$ should be smaller than the condition number of K_θ, and the forward multiplication $P_\theta x$ should be fast, ideally less than $O(n^2)$ (Meurant 1984).

Within the preconditioned conjugate gradient algorithm, we use circulant embedding to speed up multiplication of the covariance matrix by vectors. To see how circulant embedding is applicable in this nonstationary case, we write matrix-vector multiplication as follows

$$K_\theta x = C_0 x + \sum_{h=1}^{q} \omega_h \circ [C_h(\omega_h \circ x)],$$

where $\circ$ is elementwise vector multiplication, and $\omega_h = [\omega_h(s_1), \ldots, \omega_h(s_n)]^T$ is a vector of weights. In other words, to multiply K_θ by x, we need to matrix-multiply each C_h by the elementwise product of ω_h and x, perform an additional elementwise product, and sum the results. Since each C_h matrix arises from a stationary process on a grid, we can use circulant embedding to perform the required matrix multiplications in $O(n \log n)$ time. We further accelerate computation by approximating the circulant embedding with expansion factor $\frac{5}{4}$ (Guinness and Fuentes 2017). We apply this same technique to the design of P_θ. Note that if the field is not defined on a subset of a grid, circulant embedding is not typically an option. However, if a finer grid can well approximate the data locations, circulant embedding could still be potentially implemented. We explore this idea further in our analysis of temperature data in Section 6. Alternatively, methods of Chen, Wang, and Anitescu (2014) could be implemented in order to retain $O(n \log(n))$ matrix-vector multiplication.

Anitescu, Chen, and Wang (2012) fit stationary models and explore preconditioners based on the inverse of the spectral density. We extend their preconditioner development to our nonstationary covariance case by considering four different preconditioners. Letting $\gamma_1, \ldots, \gamma_n$ be the Fourier frequencies, all of the preconditioners have the general form

$$P_\theta(s_1, s_2) = \sum_{\ell=1}^{n} g(\gamma_\ell, s_1) e^{-i(s_2-s_1)'\gamma_\ell},$$

with each candidate preconditioner using a different choice for $g(\gamma, s_1)$. Namely, we consider

$$g_1(\gamma, s_1) = \frac{1}{f_0(\gamma)},$$
$$g_2(\gamma, s_1) = \sum_{h=1}^{q} \omega_h(s_1) \frac{1}{f_h(\gamma)},$$
$$g_3(\gamma, s_1) = \frac{1}{f_0(\gamma)} + \sum_{h=1}^{q} \omega_h(s_1) \frac{1}{f_h(\gamma)},$$
$$g_4(\gamma, s_1) = \frac{1}{\frac{1}{q}\sum_{h=1}^{q}[f_0(\gamma) + f_h(\gamma)]}.$$

The multiplication $P_\theta x$ is $O(n \log n)$ for these choices. For example, define G_h to be the covariance matrix formed using $1/f_h(\gamma)$ as the spectral density. Under g_3, the preconditioner multiplication can be written as

$$P_\theta x = G_0 x + \sum_{h=1}^{q} \omega_h \circ [G_h(\omega_h \circ x)],$$

which, due to the fact that each G_h is a submatrix of a circulant matrix, can be computed quickly using several FFTs, analogously to how $K_\theta x$ can be computed. Guinness and Stein (2013) explore similar preconditioners under a different nonstationary model for time series data.

To test these preconditioners, we simulate data under a model with a three-block partition and parameter settings in Table 1. Example simulations for these settings are shown in Figure 2. For each simulated data vector y, we solve the linear system $K_\theta x = y$ using each of the preconditioners. We record the time to convergence within a set tolerance of 1e−4 using the various methods for a sample size of 20,000 data points with a 200×100 data matrix for 500 replications. We give a zero vector as the starting value for all algorithms, but expect the algorithms to perform much better in practice since we use the solution from the previous step as the starting value. All preconditioners were effective at significantly reducing the convergence time and

Table 1. Simulation parameter settings.

	Z_0		Z_1 (left)		Z_2 (middle)		Z_3 (right)	
	σ^2	ρ	σ^2	ρ	σ^2	ρ	σ^2	ρ
a	0.5	1.0	1.0	1.0	1.0	3.0	0.3	0.3
b	0.5	1.0	1.0	1.0	3.0	5.0	0.3	0.3
c	0.5	1.0	1.0	1.0	5.0	7.0	0.3	0.3

NOTE: $\nu = 0.5$ and $\tau = 0$ in all cases. Sample draws for these settings are in Figure 2.

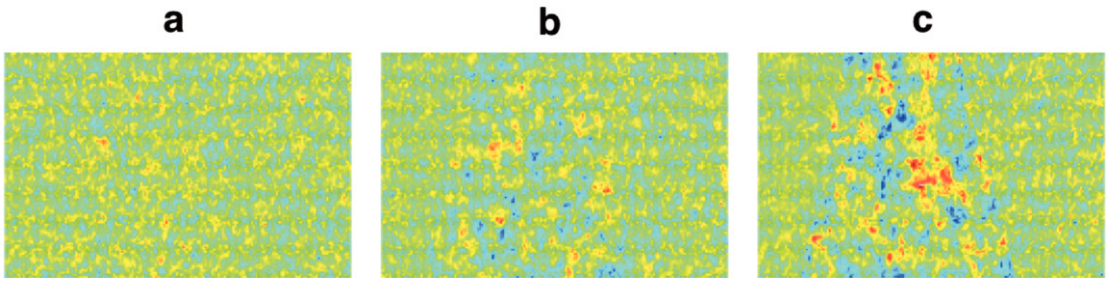

Figure 2. Sample draws from nonstationary models with parameter settings in Table 1.

Table 2. Mean total time in seconds and median iterations until convergence of conjugate gradient solve using various preconditioner matrices.

	a		b		c	
Preconditioner	Time	Iters	Time	Iters	Time	Iters
Not conditioned	0.80 (0.001)	59	1.64 (0.003)	118	2.29 (0.005)	164
Stat (g_1)	0.26 (0.000)	14	0.61 (0.001)	34	0.89 (0.001)	49
Block Stat (g_2)	0.47 (0.003)	18	0.57 (0.001)	22	0.72 (0.004)	28
Combo (g_3)	0.29 (0.001)	10	0.62 (0.003)	21	0.88 (0.002)	30
Weighted Stat (g_4)	0.39 (0.001)	16	1.09 (0.003)	46	1.82 (0.004)	77

NOTE: Convergence is defined as the square of the L^2-norm of the error vector less than 1e-4. Parentheses are standard error of the mean time estimate and the standard error of the median iterations until algorithm convergence is omitted since it is less than 0.006 in all reported cases.

number of iterations under all tested parameter settings. We choose to implement preconditioner g_2 since it is the fastest in both simulation settings b and c, but other choices may be selected and should theoretically only impact the speed of the algorithm (Table 2).

3.2. Differentiated Circulant Embedding

Multiplication of the $n \times n$ partial derivative matrix $K_\theta^{(r)}$ by a vector must also be computationally efficient. We can write the elements of the partial derivative matrix as

$$K_\theta^{(r)}(s_1, s_2) = \sum_{h=0}^{q} \omega_h(s_1)\omega_h(s_2) \int_{[0,2\pi]^2} e^{i(s_2-s_1)'\gamma} f_h^{(r)}(\gamma) d\gamma,$$

where $f_h^{(r)}$ is the partial derivative of f_h with respect to the r^{th} parameter in θ for $r = 1, 2, \ldots n_p$. This implies that $K_\theta^{(r)}$ is also a weighted sum of $n \times n$ circulant matrices, and thus, we are able to multiply it by a vector using circulant embedding. Note that since each parameter in θ appears in only one of the $(q+1)$ spectral densities f_h, only one term in the sum is nonzero for each r. We include the derivatives of the quasi-Matérn spectral density in Appendix B, supplementary materials.

4. Prediction

Prediction is a common objective in the analysis of environmental data. For variables such as temperature or air quality measures, this can mean observing data on a subset of a grid and predicting on a fine grid. These gridded predictions are then used in other analyses or as initial conditions for other mathematical or statistical models. However, kriging to a large number of prediction locations is typically computationally expensive. This section describes how one can compute these predictions efficiently under our model after estimating θ.

Recall that Y_0 is the vector of de-trended observed data collected on a subset of a regular grid, and K_θ is the covariance matrix for Y_0. Let Y_{0p} be the vector of missing de-trended values on the grid, so that $[Y_0^T, Y_{0p}^T]^T$ is a complete set of values. Define $K_{p\theta} = E(Y_{0p}Y_0^T)$ to be a matrix with dimensions of the number of predictions locations by n, and $K_{pp} = E(Y_{0p}Y_{0p}^T)$, which is a square matrix with dimensions of the number of predictions. Then the kriging predictor is $\hat{Y}_{0p} = K_{p\theta}K_\theta^{-1}Y_0$. This matrix product can be embedded in the larger system

$$\begin{bmatrix} K_\theta & K_{p\theta}^T \\ K_{p\theta} & K_{pp} \end{bmatrix} \begin{bmatrix} K_\theta^{-1}Y_0 \\ 0 \end{bmatrix} = \begin{bmatrix} Y_0 \\ \hat{Y}_{0p} \end{bmatrix}. \quad (10)$$

After computing $K_\theta^{-1}Y_0$ as outlined in Section 3.1, we perform the forward multiplication in Equation (10) to obtain the predicted values $\hat{Y}_{0p}$. The forward multiplication can be computed efficiently with circulant embedding. In the case of a large number of prediction locations, formation of the covariance matrix of the predictions $(K_{pp} - K_{p\theta}K_\theta^{-1}K_{p\theta}^T)$ is rarely possible due to memory constraints. However, a similar tool with the application of elementary matrices may be used to extract the marginal prediction variances without matrix formation. We implement these computationally efficient methods for prediction in Section 6.

5. Simulations

We performed two simulations to explore our two-step estimation procedure. The first simulation assumes that the partition is known; it compares the stochastic score parameter estimation procedure to Vecchia's approximation. In the second, we estimated both the partition and the covariance parameters. Comparison to other computationally efficient nonstationary methods is performed on real data in Section 6.

In both simulation studies, we generated data from a three-block model at spatial locations s_i for $i = 1, 2, \ldots, n$ as

$$Y(s_i) = \sum_{h=0}^{3} \omega_h(s_i) Z_h(s_i),$$

where $Z_h \overset{\text{ind}}{\sim} GP(\mu_h, C_h)$, and C_h are from the quasi-Matérn family. Let $\mu_h(s_i) = 0$ for $h = 1, 2, 3$ and $i = 1, 2, \ldots, n$. We considered the same three covariance parameter settings that were given in Table 1. The same three-block partition, seen in Figure 4, was used for each of the three covariance models. Sample draws from each of the three models are shown in Figure 2. All timings were obtained using an Intel i7- 6700HQ with 16 GB of DDR3 RAM running Windows 10 in the 64 bit program R version 3.4.3 with Microsoft R Open.

5.1. Known Partition Simulation

In this simulation, we assume knowledge of the partition and evaluate the accuracy and timing of the stochastic score method of parameter estimation. Additionally, we fix $\mu_0(s_i)$ to be 0 for $i = 1, 2, \ldots n$ in this simulation study.

We consider Vecchia's likelihood as a competitor (Vecchia 1988). Joint distributions can always be written as a product of an ordered sequence of conditional distributions. Vecchia's likelihood replaces the variables in the conditioning sets with a subset in order to reduce computational burden. Define $Y_0(s_i)$ as in Equation (6) for $i = 1, 2, \ldots, n$. Then the approximate likelihood we consider is

$$\begin{aligned} &L(\theta, Y_0(s_1), Y_0(s_2), \ldots, Y_0(s_n)|D) \\ &\quad \approx L(\theta, Y_0(s_1)|D) \prod_{i=2}^{n} L(\theta, Y_0(s_i)|Y_0(S_i), D), \end{aligned}$$

where S_i is the set of 30 nearest neighbors to the spatial location s_i in the conditioning set (Vecchia 1988).

We generate multivariate normal data with zero mean as pictured in Figure 2. We consider varying grid sizes of 20×40, 26×52, 30×60, 36×72, and 40×80 corresponding to sample sizes $n = 800; 1352; 1800; 2592;$ and 3200. We compare these results under 50 simulation replicates. In addition to varying sample size, we explore how many U_j vectors (N) are necessary for sufficient approximation to the score function. As the number of vectors increases, we know that the approximation to the score becomes more accurate, but the computing time is slowed. Thus, for each of the settings above, we consider our stochastic score method for $N = 1$, $N = 5$, and $N = 20$.

Mean squared error is a poor criterion for estimation evaluation in this case since dissimilar covariance parameter values can produce similar covariance functions. Therefore, as a criterion for evaluation of the accuracy, we define mean log-likelihood gain as

$$\frac{1}{50} \sum_{k=1}^{50} \left[2 \log L(\hat{\theta}^k, Y^k) - 2 \log L(\theta, Y^k) \right],$$

where L is the multivariate normal likelihood, θ are the true parameters, $\hat{\theta}^k$ are the estimated parameters in the kth simulation iteration, and Y^k is the response vector for the kth simulation for simulation replicates $k = 1, 2, \ldots, 50$. Note that because θ is not in fact the maximum likelihood parameters, there is likelihood to be gained by even approximate estimation, and thus, the likelihood gain is potentially positive, and a larger value is preferred. Define mean log-likelihood loss as the opposite of mean log-likelihood gain or more precisely $-1\times$(mean log-likelihood gain) (Table 3).

As sample size increases, estimation using the Vecchia likelihood approximation has a greater mean log-likelihood loss. This loss is greater, with a larger variance for setting c than settings a and b. These results can be seen in Figure 3. Note that although the mean log-likelihood loss increases as the sample size increases, the magnitude of the maximized multivariate normal log-likelihood tends to increase with sample size, in general. In contrast, the stochastic score methods' mean log-likelihood gains are positive for all tested sample sizes, which is an indication of successful maximization of the likelihood in optimization. Mean log-likelihood gain results from $N = 5$ and $N = 20$ have overlapping empirical confidence intervals computed from simulation iterations in all tested simulation settings. In a few cases, these simulation results have nonoverlapping empirical confidence intervals with the results of $N = 1$, meaning estimation with $N = 5$ or 20 is more accurate than estimation with $N = 1$. Therefore, we conclude that $N = 5$ vectors is sufficient for estimation, and we include settings for only $N = 1$ and $N = 5$ in the next simulation.

In terms of computation time, the stochastic score estimation method also outperforms estimation with the Vecchia likelihood in all cases for $N = 1$ and $N = 5$. Even in the relatively

Table 3. Mean time in seconds until convergence of estimates in various methods and sample sizes in simulation.

	Method	n = 800	n = 1352	n = 1800	n = 2592	n = 3200
a	Score $N = 1$	7.6(0.6)	12.1(0.9)	15.2(1.0)	19.5(1.4)	25.7(2.2)
	Score $N = 5$	14.6(1.1)	24.7(1.7)	30.5(1.6)	39.1(2.1)	57.3(3.8)
	Score $N = 20$	45.4(2.5)	81.8(5.4)	96.9(4.7)	123.0(5.2)	173.2(10.8)
	Vecchia App.	113.4(4.2)	211.2(6.0)	288.4(11.4)	451.0(16.1)	604.2(27.0)
b	Score $N = 1$	20.4(5.0)	36.9(4.7)	31.9(2.2)	62.8(8.5)	69.8(8.4)
	Score $N = 5$	33.3(8.7)	44.2(3.9)	90.1(18.9)	104.3(10.3)	129.2(18.1)
	Score $N = 20$	74.9(4.0)	142.5(12.8)	236.5(26.2)	299.1(29.9)	414.4(64.0)
	Vecchia App.	111.7(4.9)	194.9(8.2)	263.0(10.9)	385.6(16.6)	476.4(13.6)
c	Score $N = 1$	25.4(5.5)	28.7(2.9)	51.6(11.9)	66.0(14.1)	98.5(21.7)
	Score $N = 5$	28.3(2.4)	54.3(4.1)	66.0(3.5)	89.3(4.3)	133.5(9.4)
	Score $N = 20$	87.2(9.0)	171.1(23.6)	199.8(12.4)	276.9(16.1)	378.6(22.0)
	Vecchia App	128.5(4.7)	227.4(8.1)	315.0(10.8)	465.7(16.8)	627.6(20.4)

NOTE: In parentheses is the simulation standard error. Settings (a,b,c) refer to parameter settings seen in Table 1 and Figure 2.

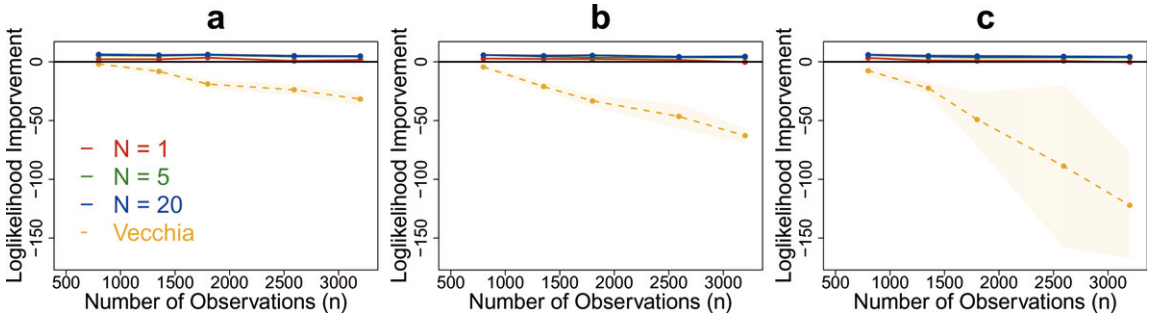

Figure 3. Mean log-likelihood gain under nonstationary settings (a,b,c) in comparison to Vecchia likelihood approximation. The shaded regions represent 95% pointwise confidence intervals in simulation.

small sample size of 3200, it takes just over 10 min on average for the Vecchia likelihood to converge in nonstationary setting a, but only about 25 sec for the stochastic score approximation with 1 vector to reach a more accurate solution. This approximately 24 times speed up is significant even in small samples, and offers essential time savings in large samples. Since we need to perform roughly five times as many linear solves in $N = 5$ as compared to $N = 1$, we would expect time to scale as such. However, the computation time for the solution with $N = 5$ is generally less than two times as long as the time for $N = 1$. This implies that each step taken with more vectors actually moves closer to the solution, and therefore, fewer iterations are needed. Hence, we conclude that especially in larger samples, the stochastic score approximation is more accurate and faster than the Vecchia approximation at estimating the maximum likelihood parameters in our nonstationary model.

5.2. Unknown Partition Simulation

In this simulation study, we evaluate the loss in log-likelihood using our proposed methods when the partition and parameters are both unknown. We again generate data with parameters in Table 1, but also allow for nonzero μ_0. Define for $i = 1, 2, \ldots n$

$$\mu_0(s_i) = X(s_i)\beta,$$

where $\beta = [0, 1]^T$ and $X(s_i) = [1, N(1, 1)]$. We allow for estimation of this nonconstant, linear mean function that is simultaneously estimated with the covariance parameters. Alternating with steps in the direction of the gradient of the covariance parameters, at step m of the parameter estimation algorithm, the current estimate of β is $\beta^{(m)}$, where m is an integer. This is updated by its generalized least squares solution

$$\beta^{(m)} = (X^T K_{\theta^{(m)}}^{-1} X)^{-1} X^T K_{\theta^{(m)}}^{-1} Y,$$

where $Y^T = [Y(s_1), Y(s_2), \ldots, Y(s_n)]$, $X^T = [X(s_1)^T, X(s_2)^T, \ldots, X(s_n)^T]$, and $\theta^{(m)}$ is the estimate of θ at the mth stage in the estimation algorithm.

The base partition has blocks B_i of size 10×10 individual locations. We consider the significance levels for the likelihood ratio tests as (0.0005, 0.001, 0.002, 0.003, 0.004, 0.005, 0.006, 0.007, 0.008, 0.009), with three random orderings of the neighboring base partition blocks for each significance level. This produces 30 candidate partitions, from which we select the best partition using BIC. Given our selected partition, we apply the stochastic score method with $N = 1$ and $N = 5$ U_j vectors. To evaluate the necessity of partition estimation, we also use the stochastic score method to estimate parameters using a naive partition that divides the domain into two blocks of equal size. We explore the accuracy of the estimation over the varying sample sizes of $n = 5000$, $n = 9800$, and $n = 16{,}200$ (Figure 4).

We evaluate the accuracy of our estimated partition with the Rand index (Rand 1971). This criterion evaluates the accuracy of clustering, and it is applicable here since a partition is a spatially-contiguous clustering of the domain. The Rand index is defined as the proportion of observations that are correctly clustered as in the true model partition,

$$\text{Rand Index} = \frac{1}{\binom{n}{2}} \sum_{i=1}^{n} \sum_{j<i} I\{t_{ij} = \widehat{t_{ij}}\},$$

where $\binom{n}{2}$ is the total number of pairs of observations, t_{ij} is an indicator that is 1 when s_i and s_j are in the same block of the true partition, and $\widehat{t_{ij}}$ is the analogous indicator for the estimated partition. Therefore, $I\{t_{ij} = \widehat{t_{ij}}\}$ is an indicator function that is 1 when the two partitions agree and 0 otherwise. We again use likelihood gain as a measure of parameter estimation as in the last simulation (Table 4).

The automatic partition selection methods always produce better results than partitioning the domain into two equal blocks as seen in Figure 5. This arbitrary partitioning is meant to represent partition selection in Fuentes (2001). So while we expect the performance of partition selection to depend on the parameters of the model and significance levels chosen, we expect the methods to outperform current partitioning methods. Although unrestricted, the algorithm selects the correct number of partition blocks in the majority of simulation settings. In simulation setting c, where the parameters are more different between the blocks, the partition selection method produces partitions with a higher mean Rand index with a smaller variance. This difference among settings is smaller in the larger sample sizes as the partitioning procedure performs well in all simulation settings when there is a large sample size. With the base partition we selected, the closest possible partition to the truth has a maximum Rand index of 0.96. One example of a selected partition can be seen in Figure 4.

Because we estimate both the partition and the covariance parameters, the mean log-likelihood gain is negative for all simulation settings. However, this loss is small relative to the sample size and magnitude of the true log-likelihood. In these larger

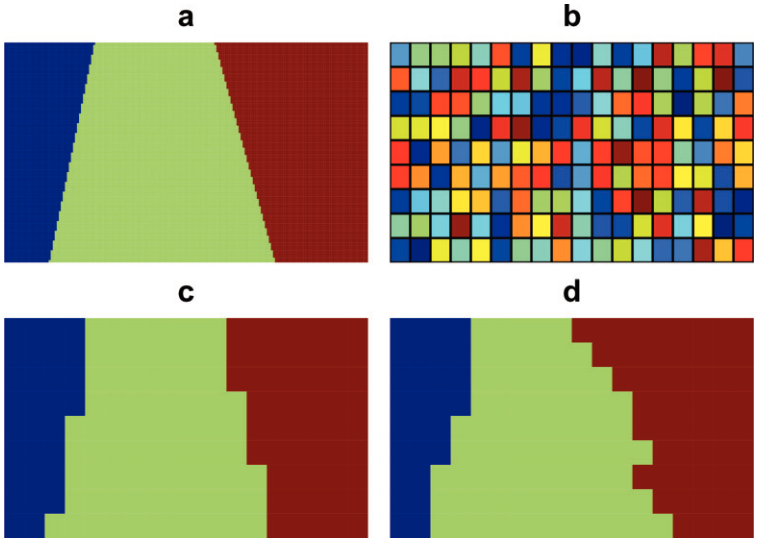

Figure 4. a. True partition, b. base partition, c. best possible partition with assumed base partition (Rand = 0.96), d. minimum BIC partition from simulation for setting 3c (Rand = 0.92).

Table 4. Results from the unknown partition simulation that include proportion of simulation iterates that correctly identified the partition to have 3 blocks, log-likelihood gain as previously defined for three implementations of the stochastic score estimation, and finally, the mean true log-likelihood value $2 \log L(\theta, Y)$, where the true simulation parameter settings θ and partition are used.

	n	Prop 3 blocks	Like gain N = 1	Like gain N = 5	Like gain eq part	True like
	5000	0.78	−66.7(2.5)	−65.2(2.6)	−139.1(3.0)	3972.6(11.8)
a	9800	0.92	−129.8(4.9)	−127.4(4.8)	−240.9(5.9)	7772.0(17.8)
	16,200	0.92	−193.0(9.5)	−190.2(9.2)	−344.3(9.0)	12805.4(25.8)
	5000	0.94	−112.0(4.3)	−108.6(4.3)	−261.8(4.6)	4860.4(15.7)
b	9800	0.84	−188.1(7.1)	−184.4(6.9)	−425.6(7.2)	9443.9(22.2)
	16,200	0.76	−287.8(12.8)	−282.9(13.1)	−602.7(12.5)	15554.0(24.5)
	5000	0.96	−152.3(4.8)	−149.2(4.6)	−421.9(6.6)	5160.2(16.1)
c	9800	0.82	−246.7(6.6)	−243.3(6.6)	−632.2(7.9)	10020.8(19.1)
	16,200	0.64	−364.7(11.7)	−354.9(10.2)	−895.6(11.9)	16574.4(21.0)

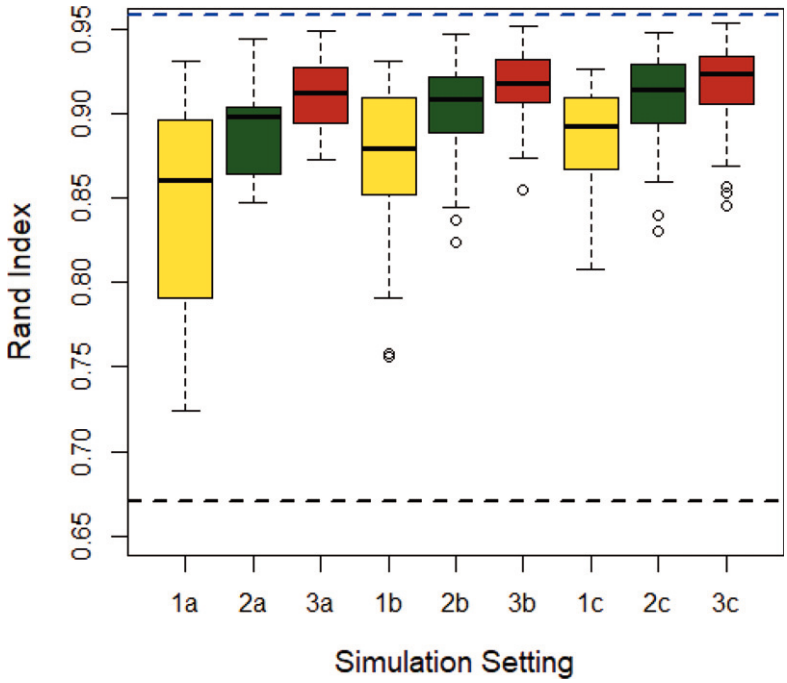

Figure 5. Boxplot of Rand indexes for sample sizes (1: $n = 5000$, 2: $n = 9800$, 3: $n = 16{,}200$) and nonstationary settings (a,b,c) as defined in Table 1 and Figure 2. The top dotted line represents the maximum possible Rand index given the base partition for the largest sample size (3), and the bottom dotted line represents Rand index for partitioning the domain into 2 equal blocks.

sample size settings, the difference in the mean log-likelihood loss in the stochastic score method using $N = 1$ and $N = 5$ is small. However, there is an empirically significant benefit to estimating the partition as opposed to equally dividing the domain. Additionally, in all tested simulations settings, empirical 95% confidence intervals in simulation iterations contain the true mean parameters with interval lengths about 0.01. Hence, we have shown in simulation the accuracy and computational efficiency of our estimation method when the partition is estimated from the data.

6. Application to Daily Temperature Data

We applied the methods developed in this manuscript to average daily temperatures from monitoring locations across the United States in 2018. Temperature is an important environmental variable that is of primary interest to track the progress of climate change (Folland et al. 2001). Additionally, temperature, particularly extreme temperatures, can impact human and environmental health (McMichael, Woodruff, and Hales 2006). Finally, temperature is needed in many numerical models in order to simulate or predict other environmental variables of interest (Byun and Schere 2006). Regardless of the interest in temperature as a variable, all of these analyses require the accurate prediction of temperature at locations without monitoring stations. Therefore, we implemented the methods presented in this article to predict temperature at finely gridded locations across the United States.

Our data is sourced from the National Oceanic and Atmospheric Administration (NOAA) and National Centers for Environmental Information's (NCEI) Global Historical Climatology Network Daily (GHCN-D) (Menne et al. 2012). This data source is comprised of over 20 monitoring networks across the world including WBAN, US Cooperative Summary of the Day, and other monitoring systems. The data are updated daily and reconstructed weekly to ensure comparability across data sources. We extracted average daily temperature in the contiguous USA for January 2, 2018, but a similar analysis may be performed on any day of interest. There are average daily temperature values for 2050 irregularly spaced monitoring locations. Further information and access to this dataset can be obtained here: https://registry.opendata.aws/noaa-ghcn/.

Because our method requires locations on a subset of a grid, we mapped locations to the closest point on a 160×240 grid in an equal area coordinate system. In the center of our domain, the grid boxes have side lengths approximately 0.2 degrees latitude and longitude. The mapping sometimes results in multiple datapoints being assigned to the same location. When this is the case, we averaged the temperature values and assigned only one point to the gridded location. The gridded dataset has 1773 observations, and the comparison of the original data and the gridded approximate locations can be seen in Figure 6.

We first performed 5-fold cross-validation to compare predictions from our methodology to those from some popular competing models. Training data is assigned using a simple random sample to one of the five possible folds. Using the methods presented in this article, we defined models $l = 1, 2, \ldots, 5$ where data from fold l is excluded. Let $Y_l(s_1)$ be the average daily temperature at location s_1 in model l. Then we modeled

$$Y_l(s_1) = Z_{0l}(s_1) + \sum_{h=1}^{q_l} \omega_{hl}(s_1) Z_{hl}(s_1), \tag{11}$$

where $Z_{0l} \sim GP(\beta_{0l}, C_{0l})$, where β_{0l} is a constant intercept parameter, and $Z_{hl} \sim GP(0, C_{hl})$ for $h = 1, 2, \ldots q_l$. We define C_{0l} and C_{hl} as in Equation (8) with the quasi-Matérn spectral densities. We included no covariates in this method or in the competing methods to compare prediction only with the respective nonstationary covariances. We implemented the stochastic score approximation estimation method outlined in the previous sections for $N = 5$ and predicted data at the omitted locations in each fold. Prediction locations were assigned to the partition block assignment of the closest observed location.

In the implementation of our partition selection method, we generated 30 candidate partitions with base partition configurations seen in Figure 7. We used significance levels in the interval $[\frac{0.05}{n_B}, .01]$, where $n_B = 24$ or 96, depending on the base partition. We found that using this significance range produces a diverse set of partitions with varying number of blocks. The best partition was chosen via BIC for each fold l.

Since this partitioning method of weightings causes nonsmooth covariances and predictions, we also considered smoothing the weights of selected partitions for parameter estimation. The smoothed weight function at spatial location s_1 is defined as a convolution of the weights with a smoothing kernel

$$\overline{\omega_h(s_1)} = \sum_{i=1}^{n} \omega_h(s_i) f(s_1 - s_i). \tag{12}$$

We took f to be a Gaussian kernel with range 0.3 units in the equal area coordinate system, and we used the FFT to compute

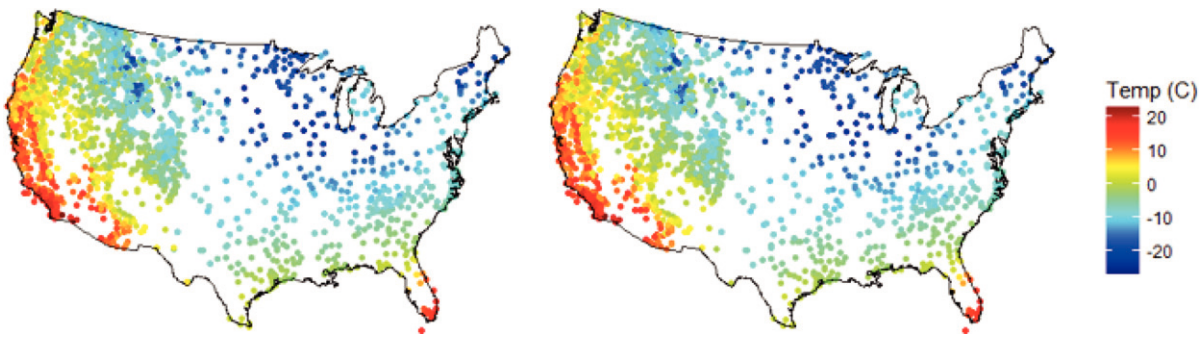

Figure 6. Original irregularly spaced average temperature (left) and data approximated to a fine grid (right).

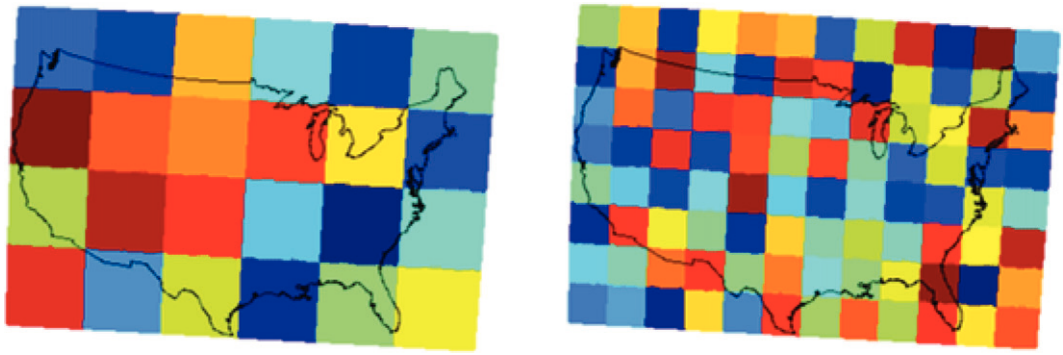

Figure 7. Base partitions used in cross-validation study. Pictured are 4 × 6 (left) and 8 × 12 (right) base partitions

Figure 8. Nonsmooth (left) and Smoothed (right) example weighting functions for a single partition block.

Table 5. Results of 5-fold cross-validation for various models and estimation methods.

Method	Base Partition	RMSE	Coverage (95% PI)	Length of PI	Time (mins)
Lattice Kriging	–	2.90	0.45	2.8	2.1
Treed GP	–	2.79	0.94	10.6	756.3
FRK	–	2.99	0.78	7.2	0.3
Stochastic Score	4 × 6	2.70	0.88	8.3	183.6
	8 × 12	2.67	0.87	8.1	224.2
	Smooth 4 × 6	2.78	0.91	10.2	221.1
	Smooth 8 × 12	2.78	0.90	9.8	278.6

the convolutions quickly. An example of these smoothed weightings is pictured in Figure 8.

For comparison, we include results from several different applicable models. First, we considered Fixed Rank Kriging (Cressie and Johannesson 2008). In this model, the spatial component is composed of a linear combination of basis functions. We implemented this method with the package "FRK" in R with two resolutions of Gaussian basis functions (Zammit-Mangion and Cressie 2017). We also compare predictions produced from Lattice Kriging (Nychka et al. 2015). We implemented this method with the package "LatticeKrig" with default settings in R (Nychka et al. 2016). Most similar to our approach is the Treed Gaussian Process (Gramacy and Lee 2008). In this Bayesian modeling averaging estimation method, the data are partitioned in a treed structure with an independent model in each block. We implemented this method with the package "tgp" in R with constant mean and otherwise default settings (Gramacy 2007; Gramacy and Taddy 2010). Results of the 5-fold cross-validation can be seen in Table 5. The statistics reported are computed averaged over the 1773 site locations so that RMSE $= \sqrt{\frac{1}{1773}\sum_{i=1}^{1773}\left[Y(s_i) - \hat{Y}(s_i)\right]^2}$, coverage $= \frac{1}{1773}\sum_{i=1}^{1773} I\{\hat{b}_i < Y(s_i) < \hat{u}_i\}$, length PI $= \frac{1}{1773}\sum_{i=1}^{1773}(\hat{u}_i - \hat{b}_i)$, where $\hat{Y}(s_i)$ is the predicted temperature from the appropriate fold model so $\hat{Y}(s_i) = \sum_{l=1}^{5} \hat{Y}_l(s_i) I\{s_i \in \text{fold}_l\}$, the estimated prediction interval for the ith site is $(\hat{b}_i, \hat{u}_i)$, and $I\{.\}$ is an indicator function that is 1 when the statement is true and 0 otherwise.

Lattice Kriging and Fixed Rank Kriging (FRK) are very fast methods, but their RMSE values are higher than the other estimation methods. The lowest RMSE was produced from our stochastic score methods without weight smoothing. However, the coverage is lower than the desired confidence level. The RMSE for the stochastic score with smoothed weights is very similar to that from Treed Gaussian Process, but the computation time for all of the stochastic score methods are lower. Since we value a smooth predictive map and more accurate uncertainty quantification, we implemented the stochastic score method with smoothed weights based on the 8 × 12 base partition for our final demonstration.

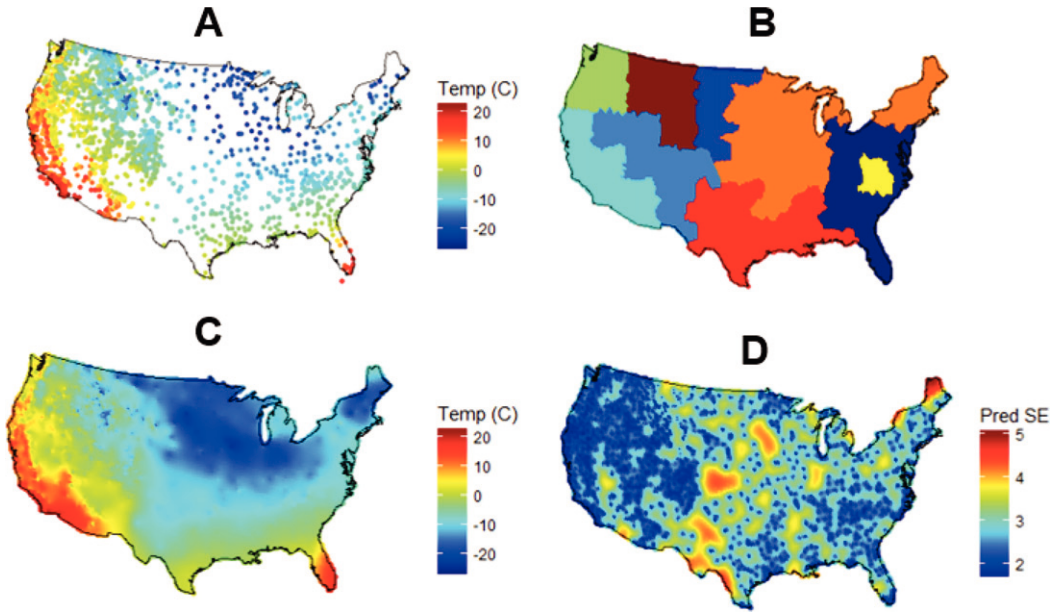

Figure 9. Results of analysis of average daily temperature on January 2, 2018. A. Data approximated to a 160 × 240 grid, B. selected partition before smoothing, where the unobserved locations are assigned to the block of the closest observed location, C. predictions, D. marginal prediction variance.

We then used the model in Equation (11) with the entire observed dataset to estimate predictions over finely gridded locations for January 2, 2018. We estimated partitions and modeled the data as described in the 5-fold cross-validation study with all 1773 observations. Because of the methods presented in Section 4, prediction to a large number of locations on grid is computationally efficient. We predicted the 36,627 unobserved locations on a 160 × 240 grid, which took approximately 39 sec on our hardware. Pictorial results for locations within the United States can be seen in Figure 9.

7. Conclusion

In this article, we have presented a new estimation method for a nonstationary model represented as the weighted linear combination of locally stationary processes and a globally stationary process. The biggest hurdle in such a model is often the definitions of the locally stationary processes, and we offered an algorithm that uses the data to estimate an irregularly shaped partition of the domain. Given any partition candidate, we estimated the parameters in large datasets on subsets of grids with n observations in only $O(n \log n)$ computational complexity per each step of estimation and $O(n)$ storage. This method of parameter estimation generalized the use of the stochastic score approximation to this nonstationary model. Within our estimation method, we proposed a matrix-free preconditioner, which reduced computing time for a linear solve of the covariance matrix by approximately 3 times for all tested settings. We generalized circulant embedding to this nonstationary case, and described a method for matrix-free prediction.

In simulation, we have shown that our estimation method with one approximation vector ($N = 1$) is more accurate and up to 24 times faster than using the Vecchia Likelihood Approximation (Vecchia 1988) to estimate parameters of the same nonstationary covariance. Finally, by applying our method to a subset of gridded temperature data, we concluded that our method was more accurate at prediction in both RMSE and coverage than the fast methods Fixed Rank Kriging (Cressie and Johannesson 2008) and Lattice Kriging (Nychka et al. 2015). With several implementations of our model, we demonstrated RMSEs lower than or similar to the RMSE in prediction using a Treed Gaussian Process (Gramacy and Lee 2008). However, the coverages of our methods were lower than that of the Treed Gaussian Process but significantly faster in computation, with better than a 63% reduction in computation time. We used our method with smoothed weighting functions to create a map of finely gridded average temperatures for January 2, 2018.

This estimation method could be adapted to a multivariate or spatio-temporal framework, which would involve further work on defining multivariate or spatial-temporal nonstationary models and proper extensions of the stochastic score for estimation. Alternatively, many large datasets have irregularly spaced observations. Further work could investigate the effects of mapping irregularly spaced locations to a fine grid as we have done in our data analysis. Although our partition selection method is more flexible than previous partitioning methods, future research could improve its performance. Random base partitioning could achieve more flexible candidate partitions. Statistical design of the tested significance levels, base partitioning, and ordering of the neighbor pairs could further improve the class of candidate partitions. Additionally, in other applications, it may be of interest to also partition the domain by spatially varying regression coefficients. Then, a linear mean function could be included in the likelihood maximization and testing of the neighboring base partitions. Finally, the uncertainty quantification of the parameters does not account for the uncertainty in the selected partition. Bayesian model averaging may be able to be applied to fully specify the posterior distributions of the parameters over the candidate partitions.

Funding

This material is based upon work supported by NSF Research Network on Statistics in the Atmosphere and Ocean Sciences (STATMOS) through grants DMS-1106862 and DMS-1107046 as well as NSF-DMS Grant Numbers 1406016, 1613219, and 1723158. It was also funded partially through NSF grant 570235. Research reported in this publication was supported by the National Institutes of Health under award number R01ES027892. This material was based upon work partially supported by the National Science Foundation under Grant DMS-1638521 to the Statistical and Applied Mathematical Sciences Institute. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Supplementary Materials

Appendices A-F provide further details on auxiliary details of the estimation method presented in this manuscript. These include: Appendix A compares dependent and independent sampling schemes in score approximation. Appendix B details the derivatives of the quasi-Matén spectral density. Appendix C describes the non-linear solver used for parameter estimation. Appendix D demonstrates the flexibility of partitions created via our method compared to other methods. Appendix E explores use of BIC as a proxy variable for the unknowable Rand Index by studying their correlation. Appendix F demonstrates the BIC of selected partitions plotted by the p-value cutoff used in its generation.

ORCID

Amanda Muyskens http://orcid.org/0000-0002-9787-1392
Joseph Guinness http://orcid.org/0000-0003-0564-6639

References

Anitescu, M., Chen, J., and Wang, L. (2012), "A Matrix-Free Approach for Solving the Parametric Gaussian Process Maximum Likelihood Problem," *SIAM Journal on Scientific Computing*, 34, A240–A262. [1028]

Byun, D., and Schere, K. L. (2006), "Review of the Governing Equations, Computational Algorithms, and Other Components of the Models-3 Community Multiscale Air Quality (CMAQ) Modeling System," *Applied Mechanics Reviews*, 59, 51–77. [1033]

Chen, J., Wang, L., and Anitescu, M. (2014), "A Fast Summation Tree Code for Matérn Kernel," *SIAM Journal on Scientific Computing*, 36, A289–A309. [1028]

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. [1026,1034,1035]

Datta, A., Banerjee, S., Finley, A. O., and Gelfand, A. E. (2016), "Hierarchical Nearest-Neighbor Gaussian Process Models for Large Geostatistical Datasets," *Journal of the American Statistical Association*, 111, 800–812. [1026]

Folland, C. K., Rayner, N. A., Brown, S., Smith, T., Shen, S., Parker, D., Macadam, I., Jones, P., Jones, R. N., Nicholls, N., and Sexton, D. M. H. (2001), "Global Temperature Change and its Uncertainties Since 1861," *Geophysical Research Letters*, 28, 2621–2624. [1033]

Fuentes, M. (2001), "A High Frequency Kriging Approach for Nonstationary Environmental Processes," *Environmetrics*, 12, 469–483. [1025,1026,1031]

——— (2002), "Interpolation of Nonstationary Air Pollution Processes: A Spatial Spectral Approach," *Statistical Modelling*, 2, 281–298. [1025]

Fuentes, M., Chaudhuri, A., and Holland, D. M. (2007), "Bayesian Entropy for Spatial Sampling Design of Environmental Data," *Environmental and Ecological Statistics*, 14, 323–340. [1025]

Gramacy, R. B. (2007), "tgp: An R Package for Bayesian Nonstationary, Semiparametric Nonlinear Regression and Design by Treed Gaussian Process Models," *Journal of Statistical Software*, 19, 1–46. R package version 2.4-14. [1034]

Gramacy, R. B., and Lee, H. K. H. (2008), "Bayesian Treed Gaussian Process Models with an Application to Computer Modeling," *Journal of the American Statistical Association*, 103, 1119–1130. [1026,1034,1035]

Gramacy, R. B., and Taddy, M. (2010), "Categorical Inputs, Sensitivity Analysis, Optimization and Importance Tempering with tgp Version 2, an R package for Treed Gaussian Process Models," *Journal of Statistical Software*, 33, 1–48. R package version 2.4-14. [1034]

Guinness, J., and Fuentes, M. (2015), "Likelihood Approximations for Big Nonstationary Spatial temporal Lattice Data," *Statistica Sinica*, 25, 329–349. [1026]

——— (2017), "Circulant Embedding of Approximate Covariances for Inference from Gaussian Data on Large Lattices," *Journal of Computational and Graphical Statistics*, 26, 88–97. [1027,1028]

Guinness, J., and Stein, M. L. (2013), "Transformation to Approximate Independence for Locally Stationary Gaussian Processes," *Journal of Time Series Analysis*, 34, 574–590. [1028]

Haas, T. C. (1995), "Local Prediction of a Spatio-Temporal Process with an Application to Wet Sulfate Deposition," *Journal of the American Statistical Association*, 90, 1189–1199. [1026]

Heaton, M. J., Christensen, W. F., and Terres, M. A. (2017), "Nonstationary Gaussian Process Models Using Spatial Hierarchical Clustering from Finite Differences," *Technometrics*, 59, 93–101. [1026]

Heaton, M. J., Datta, A., Finley, A. O., Furrer, R., Guinness, J., Guhaniyogi, R., Gerber, F., Gramacy, R. B., Hammerling, D., Katzfuss, M., Lindgren, F., Nychka, D. W., Sun, F., and Zammit-Mangion, A. (2018), "A Case Study Competition Among Methods for Analyzing Large Spatial Data," *Journal of Agricultural, Biological and Environmental Statistics*, 24, 398–425. [1026]

Hestenes, M. R., and Stiefel, E. (1952), *Methods of Conjugate Gradients for Solving Linear Systems* (Vol. 49), Washington, DC: NBS. [1028]

Higdon, D. (1998), "A Process-Convolution Approach to Modelling Temperatures in the North Atlantic Ocean," *Environmental and Ecological Statistics*, 173, 173–190. [1026]

Konomi, B. A., Sang, H., and Mallick, B. K. (2014), "Adaptive Bayesian Nonstationary Modeling for Large Spatial Datasets Using Covariance Approximations," *Journal of Computational and Graphical Statistics*, 23, 802–829. [1026]

McMichael, A. J., Woodruff, R. E., and Hales, S. (2006), "Climate Change and Human Health: Present and Future Risks," *The Lancet*, 367, 859–869. [1033]

Menne, M. J., Durre, I., Vose, R. S., Gleason, B. E., and Houston, T. G. (2012), "An Overview of the Global Historical Climatology Network-Daily Database," *Journal of Atmospheric and Oceanic Technology*, 29, 897–910. [1033]

Meurant, G. (1984), "The Block Preconditioned Conjugate Gradient Method on Vector Computers," *BIT Numerical Mathematics*, 24, 623–633. [1028]

Nychka, D., Bandyopadhyay, S., Hammerling, D., Lindgren, F., and Sain, S. (2015), "A Multiresolution Gaussian Process Model for the Analysis of Large Spatial Datasets," *Journal of Computational and Graphical Statistics*, 24, 579–599. [1026,1034,1035]

Nychka, D., Hammerling, D., Sain, S., and Lenssen, N. (2016), "Latticekrig: Multiresolution Kriging Based on Markov Random Fields," R package version 7.0. [1034]

Park, C., and Apley, D. (2018), "Patchwork Kriging for Large-Scale Gaussian Process Regression," *The Journal of Machine Learning Research*, 19, 269–311. [1026]

Rand, W. M. (1971), "Objective Criteria for the Evaluation of Clustering Methods," *Journal of the American Statistical Association*, 66, 846–850. [1031]

Risser, M. D., Calder, C. A., Berrocal, V. J., and Berrett, C. (2016), "Nonstationary Spatial Process Modeling via Treed Covariate Segmentation, with Application to Soil Organic Carbon Stock Assessment," arXiv preprint arXiv:1608.05655. [1025]

Sampson, P. D., and Guttorp, P. (1992), "Nonparametric Estimation of Nonstationary Spatial Covariance Structure," *Journal of the American Statistical Association*, 87, 108–119. [1026]

Stein, M. L., Chen, J., and Anitescu, M. (2013), "Stochastic Approximation of Score Functions for Gaussian Processes," *The Annals of Applied Statistics*, 7, 1162–1191. [1026,1027]

Vecchia, A. V. (1988), "Estimation and Model Identification for Continuous Spatial Processes," *Journal of the Royal Statistical Society*, Series B, 50, 297–312. [1030,1035]

Zammit-Mangion, A., and Cressie, N. (2017), "Frk: An r Package for Spatial and Spatio-Temporal Prediction with Large Datasets," R package version 0.2.2. arXiv preprint arXiv:1705.08105. [1034]