Statistically and Computationally Efficient Estimating Equations for Large Spatial Datasets

Ying Sun[1] and Michael L. Stein[2]

September 22, 2014

Abstract

For Gaussian process models, likelihood based methods are often difficult to use with large irregularly spaced spatial datasets, because exact calculations of the likelihood for n observations require $O(n^3)$ operations and $O(n^2)$ memory. Various approximation methods have been developed to address the computational difficulties. In this paper, we propose new unbiased estimating equations based on score equation approximations that are both computationally and statistically efficient. We replace the inverse covariance matrix that appears in the score equations by a sparse matrix to approximate the quadratic forms, then set the resulting quadratic forms equal to their expected values to obtain unbiased estimating equations. The sparse matrix is constructed by a sparse inverse Cholesky approach to approximate the inverse covariance matrix. The statistical efficiency of the resulting unbiased estimating equations are evaluated both in theory and by numerical studies. Our methods are applied to nearly 90,000 satellite-based measurements of water vapor levels over a region in the Southeast Pacific Ocean.

Some key words: Inverse covariance matrix; Iterative methods; Sparse matrices; Statistical efficiency; Unbiased estimating equations.

Short title: Estimating Equations for Large Spatial Datasets

[1]CEMSE Division, King Abdullah University of Science and Technology, Thuwal 23955-6900 Saudi Arabia. E-mail: ying.sun@kaust.edu.sa

[2]Department of Statistics, University of Chicago, Chicago, IL, 60637. E-mail: stein@galton.uchicago.edu.

1 Introduction

Gaussian process models are widely used in spatial statistics (Stein, 1999) and statistical analysis of computer experiments (Fang et al., 2005 and Santner et al., 2003). Gaussian process models, for example, can be used to describe the spatial variability in the process, predict unobserved values of the process and provide prediction uncertainties. Nowadays, with the dramatic increase in the capability of data collection, observational and computer-generated spatial datasets from, for example, satellite measurements or climate model outputs, often cover large regions and/or have high resolutions. Likelihood based methods, including Bayesian methods, are appropriate choices for statistical inferences on the unknown covariance structure. However, spatial processes are often observed at irregular locations, with non-negligible correlation among most observations, so that covariance matrices are often unstructured and dense. If the covariance matrix has no exploitable structure, the standard way of calculating the likelihood exactly is to compute the Cholesky decomposition of the covariance matrix, which generally requires $O(n^3)$ computations and $O(n^2)$ memory for n observations. Therefore, the computational cost is prohibitive for sufficiently large n even on large clusters of processors.

Specifically, suppose data are observed from a zero-mean, stationary and isotropic Gaussian random field Z on a domain $\mathcal{D} \subset \mathbb{R}^d$. Let $K(h;\boldsymbol{\theta})$ be the parametric covariance function between any two observations whose locations are apart by a distance h. The parameter vector $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)^T \in \mathbb{R}^p$ needs to be estimated from $\mathbf{Z} = (Z_1, \ldots, Z_n)^T$. Since the random vector $\mathbf{Z}$ follows a multivariate normal distribution, the loglikelihood up to an additive constant is given by

$$\ell(\boldsymbol{\theta}) = -\frac{1}{2}\mathbf{Z}^T\Sigma^{-1}(\boldsymbol{\theta})\mathbf{Z} - \frac{1}{2}\log|\Sigma(\boldsymbol{\theta})|, \tag{1}$$

where $\Sigma(\boldsymbol{\theta})$ is the $n \times n$ covariance matrix. Hereafter, write $\Sigma(\boldsymbol{\theta})$ as Σ for simplicity. Here, we assume the mean of the process is zero. In practice, the mean of $\mathbf{Z}$ is often assumed to be a vector that depends linearly on unknown parameters, in which case restricted maximum

likelihood (REML) should be used (Kitanidis, 1983, Stein, 1999 and Stein et al., 2004). The restricted likelihood can be written in a form that is analogous to (1) by using only contrasts, or linear combinations of the observations whose means do not depend on the unknown parameters in the mean vector. For simplicity, we will illustrate our methodology assuming the mean is known to be zero. Our methods can be adapted to the REML approach when unknown mean parameters are present.

Exact calculations of both terms in (1) require substantial computations for large n. In the quadratic term, iterative methods often provide an efficient way of computing $\Sigma^{-1}\mathbf{Z}$, where the main computation is multiplying Σ by a vector, which requires $O(n^2)$ computation per iteration for dense unstructured Σ. Iterative methods are appealing in terms of storage because, assuming elements of Σ can be computed rapidly as needed, they allow for matrix-free implementation, which does not store any $n \times n$ covariance matrix explicitly, but accesses the matrix by evaluating matrix-vector products an element at a time. Furthermore, exploitable structures of Σ, sparseness or Toeplitz, for instance, often make fast matrix-vector multiplication possible. Even for dense, unstructured Σ, iterative solvers may require many fewer ($O(mn^2)$) flops (floating-point operations) than the Cholesky decomposition ($O(n^3)$), if the number of iterations $m \ll n$. The number of iterations m for accurate solutions is related to the condition number of Σ. When nearby observations are strongly correlated, the condition number can be very large and iterative methods may require many iterations to converge, so that preconditioning is necessary to reduce m. Specifically, for two-sided preconditioning, we need to find a matrix P such that $P^T\Sigma P$ is well-conditioned and multiplying a vector by P is fast. Therefore, a good choice of preconditioner is a trade-off between the enhanced convergence and the extra cost of multiplication. If a good preconditioner is available, then the main computation in the loglikelihood is usually due to calculating $\log|\Sigma|$. See Aune et al. (2013) for some developments on stochastic approximation to log determinants of positive definite matrices.

If we consider score equations to find maximum likelihood estimates (MLEs), by setting the

gradient of the loglikelihood equal to 0, we are able to avoid the determinant computation in the likelihood. Define $\Sigma_i = \partial\Sigma(\boldsymbol{\theta})/\partial\theta_i$. The score equations for $\boldsymbol{\theta}$ are (after multiplying by a factor 2)

$$\mathbf{Z}^T\Sigma^{-1}\Sigma_i\Sigma^{-1}\mathbf{Z} - \mathrm{tr}(\Sigma^{-1}\Sigma_i) = 0, \quad i = 1, \ldots, p, \tag{2}$$

where the computation requires one solve in Σ when computing the quadratic term, but n solves in the trace term, which may not be any easier than computing $\log|\Sigma|$.

Besides exact likelihood computation, there are various approximation methods that reduce computations and/or storage, many of which have been reviewed by Sun et al. (2012), such as covariance tapering, separable covariance structures, composite likelihood based methods, spectral methods, low rank approximations and Markov models. All these methods involve approximating the likelihood itself and not the score equation; see, for example, Caragea (2003), Fuentes (2007) and Stein et al. (2004). Some methods use models that allow for efficient matrix calculations; for example, separable covariance structures reduce the dimension of the covariance matrix by ignoring the interaction between different types of covariances (Genton, 2007 and Gneiting et al., 2006); covariance tapering makes use of sparse covariance matrices by assuming the covariance function has compact support and its range is sufficiently small (Furrer et al., 2006 and Kaufman et al., 2008); and Markov models lead to sparseness of the precision matrix, the inverse of the covariance matrix, due to the property that the conditional distributions only depend on nearby neighbors (Lindgren et al., 2011 and Rue et al., 2002).

Among these approximation methods, covariance tapering is a fairly popular choice. The idea of tapering is to set the covariances at large distances to zero so that sparse matrix algorithms can be used. The tapering is done in a way such that the new tapered covariance matrix retains the property of positive definiteness and does not distort the implied local behavior of the process. For parameter estimation, Kaufman et al. (2008) showed that tapered covariance matrices for likelihood approximations can yield biased estimating equations, and proposed one way to correct

the bias. Stein (2013) introduced a new approach also based on tapered covariance matrices that sometimes yields considerably better unbiased estimating equations than those proposed in Kaufman et al. (2008). Although approximation methods may reduce computation and memory requirements, a natural question is how much statistical efficiency is lost compared to the exact solution. Stein (2013) studied the statistical properties of covariance tapers and compared the statistical efficiency of different estimating equations with tapered covariance matrices. The numerical results indicate that independent blocks, where data are simply divided into blocks and one assumes independence across blocks, are usually better than isotropic tapers, sometimes by a large margin.

In this paper, we propose three score equation approximations yielding unbiased estimating equations that can save both computations and storage. We replace one or both appearances of Σ^{-1} in the quadratic form $\mathbf{Z}^T\Sigma^{-1}\Sigma_i\Sigma^{-1}\mathbf{Z}$ by a sparse matrix, then set the resulting quadratic forms equal to their expected values to obtain unbiased estimating equations. Stein (2013) used a similar approach to reducing computations, but instead of making Σ^{-1} sparse, used tapered covariance matrices to obtain a sparse approximation of Σ, in which case, the resulting Σ^{-1} approximation is not generally sparse. Our approach is also different from Markov models even though we replace both Σ^{-1} by a sparse matrix. Suppose X is a Gaussian Markov Random Field (GMRF) with zero-mean, then the random vector $\mathbf{X} = (X_1, \ldots, X_n)^T$ follows a multivariate normal distribution, and the loglikelihood up to an additive constant is given by

$$-\frac{1}{2}\mathbf{X}^T Q\mathbf{X} + \frac{1}{2}\log|Q|,$$

where Q is the inverse covariance matrix that is sparse for $\mathbf{X}$ observed from a GMRF. Although the likelihood of the Markov model can be exactly calculated, the MLEs are not for the natural model (1) we want to fit for $\mathbf{Z}$ observed from a Gaussian random field. Since we prefer to approximate the exact MLEs from the natural model without assuming some other model that allows for exact computation, we develop different unbiased estimating equations to approximate

the natural score equations of interest, compare their computational requirements, and assess the statistical efficiency of different approximations.

The rest of our paper is organized as follows. Section 2 develops the detailed methodology. In Section 2.1, we propose three sets of new unbiased estimating equations based on approximating Σ^{-1} and discuss the computational requirements of these score equation approximations. Section 2.2 describes the parameter estimation procedure in detail. Sections 2.3 and 2.4 describe, respectively, how to construct a sparse approximation of Σ^{-1}. In Section 3.1, we numerically examine the theoretical loss in statistical efficiency of these estimating equations compared to the exact score equations. The performance of the estimation procedures are also evaluated by simulations in Section 3.2. In Section 4, the proposed methods are applied to estimate the variogram of high resolution satellite water vapor measurements. Some limitations and possible improvements of the proposed methods are discussed in Section 5. Computational details can be found in the Appendix.

2 Methodology

2.1 Unbiased Estimating Equations

A general way to obtain unbiased estimating equations is to set quadratic forms equal to their expected values, i.e., $\mathbf{Z}^T A_i \mathbf{Z} - \mathrm{tr}(A_i \Sigma) = 0$, since $E(\mathbf{Z}^T A_i \mathbf{Z}) = \mathrm{tr}(A_i \Sigma)$ for $i = 1, \ldots, p$. In fact, the resulting estimating equations are the score equations (2) when $A_i = \Sigma^{-1} \Sigma_i \Sigma^{-1}$. When it is impossible to compute the exact score equations, we propose to approximate $\Sigma^{-1} \Sigma_i \Sigma^{-1}$ in the quadratic forms of the score equations by A_i, where A_i is chosen to avoid multiple solves in Σ while retaining as much statistical efficiency as possible.

Let V be a sparse matrix, which ideally provides a good approximation to Σ^{-1}. One choice of a computationally less demanding set of unbiased estimating equations than the exact score

equations is

$$\mathbf{Z}^T V \Sigma_i \Sigma^{-1} \mathbf{Z} - \mathrm{tr}(V \Sigma_i) = 0, \quad i = 1, \ldots, p, \tag{3}$$

where the first Σ^{-1} in $\mathbf{Z}^T \Sigma^{-1} \Sigma_i \Sigma^{-1} \mathbf{Z}$ is replaced by a sparse V. Then the resulting quadratic forms require computing $\Sigma^{-1}\mathbf{Z}$, so one solve in Σ as in the score equations. However, the trace term $\mathrm{tr}(V\Sigma_i)$ avoids the n solves for calculating $\mathrm{tr}(\Sigma^{-1}\Sigma_i)$ in the score equations, requiring only elementwise matrix multiplication, which is a much easier task, especially when V is sparse.

Another choice is to replace both appearances of Σ^{-1} in $\mathbf{Z}^T \Sigma^{-1} \Sigma_i \Sigma^{-1} \mathbf{Z}$ by a sparse V, leading to the unbiased estimating equations

$$\mathbf{Z}^T V \Sigma_i V \mathbf{Z} - \mathrm{tr}(V \Sigma_i V \Sigma) = 0, \quad i = 1, \ldots, p. \tag{4}$$

Comparing to (3), no solves are involved to compute the estimating equations (4), which saves computations from the iterative method, whereas the trace term will generally require more calculations than the trace term in (3), since $\mathrm{tr}(V\Sigma_i)$ requires only the diagonal elements of $V\Sigma_i$ while $\mathrm{tr}(V\Sigma_i V\Sigma)$ requires all the elements of $V\Sigma_i$ and $V\Sigma$. Detailed operation requirements are discussed in Section 2.2.

Both (3) and (4) provide good approximations to the score equations (2) if V is a good approximation to Σ^{-1}. Since only one instance of Σ^{-1} is replaced by V in (3), it provides a better approximation than (4) does. In fact, for a sufficiently good V, we can obtain an even better score equation approximation by considering the linear combination of (3) and (4),

$$\mathbf{Z}^T \{2V\Sigma_i\Sigma^{-1} - V\Sigma_i V\}\mathbf{Z} - \mathrm{tr}\{2V\Sigma_i - V\Sigma_i V\Sigma\} = 0, \quad i = 1, \ldots, p. \tag{5}$$

To see why (5) might be better than (3) and (4), for each set of the estimating equations, compute the variance of its difference from the exact score functions. Let $\Delta = \Sigma^{-1} - V$. Since

$$\mathrm{var}\big(\mathbf{Z}^T A_i \mathbf{Z}\big) = 2\,\mathrm{tr}\left\{\left(\frac{A_i + A_i^T}{2}\Sigma\right)^2\right\},$$

for (3)−(5), we can show that

$$E_1 = \mathrm{var}\Big\{\mathbf{Z}^T(V\Sigma_i\Sigma^{-1} - \Sigma^{-1}\Sigma_i\Sigma^{-1})\mathbf{Z}\Big\} = 2\,\mathrm{tr}\Big\{\frac{1}{4}\Big(\Delta\Sigma_i + \Sigma^{-1}\Sigma_i\Delta\Sigma\Big)^2\Big\},$$
$$E_2 = \mathrm{var}\Big\{\mathbf{Z}^T(V\Sigma_iV - \Sigma^{-1}\Sigma_i\Sigma^{-1})\mathbf{Z}\Big\} = 2\,\mathrm{tr}\Big\{\Big(\Delta\Sigma_iV\Sigma + \Sigma^{-1}\Sigma_i\Delta\Sigma\Big)^2\Big\},$$
$$E_3 = \mathrm{var}\Big\{\mathbf{Z}^T(2V\Sigma_i\Sigma^{-1} - V\Sigma_iV - \Sigma^{-1}\Sigma_i\Sigma^{-1})\mathbf{Z}\Big\} = 2\,\mathrm{tr}\Big\{\Big(\Delta\Sigma_i\Delta\Sigma\Big)^2\Big\}.$$

Suppose there is a sequence of matrices V_k, such that, for $\Delta_k = \Sigma^{-1} - V_k$, $\|\Delta_k\|_F \to 0$, as $k \to \infty$, where $\|\cdot\|_F$ denotes the Frobenius norm. As $k \to \infty$, using $\|A+B\|_F \le \|A\|_F + \|B\|_F$ and $\|AB\|_F \le \|A\|_F\|B\|_F$, we have $E_{1k} = O(\|\Delta_k\|_F^2)$ and $E_{2k} = O(\|\Delta_k\|_F^2)$, but neither is generally $O(\|\Delta_k\|_F^4)$, whereas $E_{3k} = O(\|\Delta_k\|_F^4)$. Therefore, (5) provides a better approximation to the exact score equations than either (3) or (4) does for a sufficiently small $\|\Delta_k\|_F$. However, because (5) involves taking a difference of quadratic forms, it can perform worse than (3) or (4) if V is not a sufficiently good approximation to Σ^{-1}. To give an example of a sequence of V_k's for which $\|\Delta_k\|_F \to 0$, as $k \to \infty$, let λ_i, $i = 1, \ldots, n$, be the eigenvalues of Σ, and $\xi \in \mathbb{R}$ be a scale factor, such that $|1 - \xi\lambda_i| < 1$ for all i. The Taylor expansion of Σ^{-1} can be written as

$$\Sigma^{-1} = \xi\big\{I - (I - \xi\Sigma)\big\}^{-1} = \xi\sum_{j=0}^{\infty}(I - \xi\Sigma)^j,$$

and V_k can be defined as the first k terms of the Taylor expansion.

The proposed estimating equations are innovative in the sense that they allow us to take advantage of the existing iterative methods for exact computations, as well as avoid multiple solves. Other approaches for score equation approximation also exist. For example, Anitescu et al. (2012) proposed a stochastic approximation approach based on the Hutchinson trace estimator (Hutchinson, 1990). In this approach, the trace term is approximated by

$$\frac{1}{N}\sum_{j=1}^{N} U_j^T\Sigma^{-1}\Sigma_iU_j,$$

where the U_j's are i.i.d. random vectors with i.i.d. symmetric Bernoulli components. Then the

computation can be reduced to N solves, which is computationally efficient if we can take N much smaller than n. As shown in Stein et al. (2012), the value N affects the statistical efficiency and should increase as the condition number of Σ grows. Stein et al. (2012) found that even after preconditioning, N of around 100 is often needed for good statistical properties, so the stochastic approximation approach still generally requires a substantial number of solves in Σ, whereas (3) and (5) only require a single solve and (4) none at all.

2.2 Parameter Estimation

When the covariance function $\Sigma(\boldsymbol{\theta}) = \phi C(\boldsymbol{\alpha})$, where $\boldsymbol{\theta} = (\phi, \boldsymbol{\alpha})$, so that Σ is known up to a scale parameter ϕ and a vector parameter $\boldsymbol{\alpha}$, there is a simple closed form for the maximum likelihood estimate of ϕ as a function of $\boldsymbol{\alpha}$:

$$\hat{\phi}(\boldsymbol{\alpha}) = \frac{1}{n}\mathbf{Z}^T C^{-1}(\boldsymbol{\alpha})\mathbf{Z}. \tag{6}$$

We obtain profile estimating equations by plugging (6) into (3) and (4), respectively:

$$\mathbf{Z}^T V C_i C^{-1}\mathbf{Z} - \frac{1}{n}\mathbf{Z}^T C^{-1}\mathbf{Z}\,\mathrm{tr}(VC_i) = 0, \tag{7}$$

$$\mathbf{Z}^T V C_i V\mathbf{Z} - \frac{1}{n}\mathbf{Z}^T C^{-1}\mathbf{Z}\,\mathrm{tr}(VC_iVC) = 0, \tag{8}$$

and the analog of (5) becomes

$$\mathbf{Z}^T(2VC_iC^{-1} - VC_iV)\mathbf{Z} - \frac{1}{n}\mathbf{Z}^T C^{-1}\mathbf{Z}\,\mathrm{tr}(2VC_i - VC_iVC) = 0, \tag{9}$$

where $C_i = \partial C(\boldsymbol{\alpha})/\partial\alpha_i$, $i = 1, \ldots, p-1$. Note that (7)$-$(9) are all unbiased estimating equations.

For (7) and (9), iterative methods are used to find $C^{-1}\mathbf{Z}$. As explained in Section 1, when nearby observations are strongly correlated, we need to precondition in order to find the solution in just a few iterations. Let W be the preconditioner. The estimating procedure is as follows:

1. For a given value of $\boldsymbol{\alpha}$, compute the terms in the estimating equation:

(a) Construct a sparse V as an approximation to C^{-1}.

(b) Find $C^{-1}\mathbf{Z}$ by solving the preconditioned linear system $WC\mathbf{x} = W\mathbf{Z}$ using iterative methods.

2. Solve a nonlinear system (7), (8) or (9) over the parameter space of $\boldsymbol{\alpha}$, and obtain $\hat{\boldsymbol{\alpha}}$.

3. Find $\hat{\phi}$ by (6). Note that $\mathbf{Z}^T C^{-1}(\hat{\boldsymbol{\alpha}})\mathbf{Z}/n$ has already been found as part of (9) in the last iteration, so it does not require extra calculations.

There are many different ways to choose the preconditioner W (Chen, 2005). For example, $W = V$ is a natural choice if V is a good approximation to C^{-1}, although it is not essential that V be used to precondition. In Section 3.2, we briefly consider the possibility of using $W \neq V$.

In the estimating procedure, matrix-vector multiplication is required both in the iterative method and for evaluating the estimating equations. Take the calculations of $C\mathbf{x}$ for example. Writing r for the number of flops to compute $C\mathbf{x}$, the worst case is $r = O(n^2)$. When observations are on a regular grid and a stationary model is assumed, C is a Block Toeplitz Toeplitz Block (BTTB) matrix, which can be embedded in a Block Circulant Circulant Block (BCCB) matrix. In this way, memory can be saved by only storing the first row of the BCCB matrix, and the matrix-vector multiplication can be carried out by 2D Fast Fourier Transform (FFT), in which case $r = O(n \log n)$. The circulant embedding technique is still applicable when some observations on the regular grid are missing. If $\mathbf{x}$ only has $O(q)$ non-zero entries, then $r = O(nq)$ and only q entries in each row of C need to be calculated. Therefore, (7) requires $O(mn^2)$ operations, where m is the number of iterations in the iterative method; (8) requires $O(mn^2) + O(rn)$ operations, although (4) only requires $O(rn)$.

There are two iterations in the estimating procedure: solving the linear system $C\mathbf{x} = \mathbf{Z}$ for a given $\boldsymbol{\theta}$ value, and solving the nonlinear estimating equations in $\boldsymbol{\theta}$. The operation counts above are for each $\boldsymbol{\theta}$ value, and constructing a good quality of V is also expensive, especially if V depends on $\boldsymbol{\theta}$. Thus, it is worthwhile to explore how to arrange the nonlinear solver to

save computations in each updating step and to achieve convergence in fewer steps as well. For example, estimating equations (9) require more computations than (7). In addition, as discussed in Section 2.1, a sufficiently good V, which also requires more computations, is necessary for (9) to outperform (7) or (8). To save computations, one can construct a V that does not approximate Σ^{-1} so well, and choose (7) to obtain $\hat{\boldsymbol{\theta}}_7$, then start from $\hat{\boldsymbol{\theta}}_7$ in the nonlinear solver, and do one-step update using (9) with a V that better approximates Σ^{-1} to obtain an approximate MLE $\hat{\boldsymbol{\theta}}_9$. This approach is used in the application in Section 4. Furthermore, it may not be essential to update V at each $\boldsymbol{\theta}$ value, which could potentially save a fair amount of computations, although we do not explore that possibility here.

2.3 Approximating Σ^{-1}

Since any joint density can be written as a product of conditional densities based on some ordering of the observations, one way to reduce the computations is to condition on only a subset of the "past" observations when computing the conditional distribution of each observation. In this section, we show how this approach, proposed by Vecchia (1988) as a way to approximate the loglikelihood directly rather than the score equations, can be adapted to obtain a sparse approximation of Σ^{-1}. This approximation is called a sparse inverse Cholesky decomposition by Kolotilina and Yeremin (1993), who used it in the preconditioning context.

Specifically, suppose that $\mathbf{Z} = (Z_1, \ldots, Z_n)^T \sim N(\mathbf{0}, \Sigma)$. The loglikelihood can be written as

$$\log f(Z_1) + \sum_{j=2}^{n} \log f(Z_j | \mathbf{Z}_{j-1}),$$

where f is the normal density and $\mathbf{Z}_{j-1} = (Z_1, \ldots, Z_{j-1})^T$. For $j > 1$,

$$\mathrm{cov}\begin{pmatrix} \mathbf{Z}_{j-1} \\ Z_j \end{pmatrix} = \begin{pmatrix} \Sigma_{j-1} & \boldsymbol{\sigma}_{j-1} \\ \boldsymbol{\sigma}_{j-1}^T & \sigma_j \end{pmatrix},$$

and the logarithm of the conditional density, up to an additive constant, is given by

$$\log f(Z_j|\mathbf{Z}_{j-1}) = -\frac{1}{2}\frac{\left(Z_j - \boldsymbol{\sigma}_{j-1}^T\Sigma_{j-1}^{-1}\mathbf{Z}_{j-1}\right)^2}{\sigma_j - \boldsymbol{\sigma}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{\sigma}_{j-1}} - \frac{1}{2}\log\left(\sigma_j - \boldsymbol{\sigma}_{j-1}^T\Sigma_{j-1}^{-1}\boldsymbol{\sigma}_{j-1}\right),$$

which is the log-density of $W_j = \mathbf{b}_j^T\mathbf{Z} \sim N(0, V_j)$ with $V_j = \mathbf{b}_j^T\Sigma\mathbf{b}_j$. Here $\mathbf{b}_j$ is a $n \times 1$ vector given by, for $j > 1$,

$$\mathbf{b}_j = \left(-\boldsymbol{\sigma}_{j-1}^T\Sigma_{j-1}^{-1}, 1, 0, \ldots, 0\right)^T,$$

and $\mathbf{b}_1 = (1, 0, \ldots, 0)^T$ for $j = 1$. Then from the likelihood, it is easy to see that $\Sigma^{-1} = \sum_{j=1}^n \mathbf{b}_j V_j^{-1}\mathbf{b}_j^T$.

If we compute the conditional distribution of Z_j using only a subset of $\mathbf{Z}_{j-1}$, it essentially forces 0's into $\mathbf{b}_j$, and leads to a sparse V. For $j > 1$, let $\mathbf{S}_j$ be the conditioning vector which is the subvector of $\mathbf{Z}_{j-1}$ on which the conditional distribution of Z_j is based. Denote the resulting matrices and vectors by $\boldsymbol{\sigma}_{j-1}^T(\mathbf{S}_j)$, $\Sigma_{j-1}^{-1}(\mathbf{S}_j)$, $\mathbf{b}_j(\mathbf{S}_j)$, and $V_j(\mathbf{S}_j)$. A sparse V can be constructed to approximate Σ^{-1} in the following steps:

1. Order observations such that $\mathbf{Z} = (Z_1, \ldots, Z_n)^T$.

2. Choose the length s of the conditioning vector $\mathbf{S}_j$.

3. For $j \leq s$, $\mathbf{S}_j$ is $\mathbf{Z}_{j-1}$; for $j > s$, $\mathbf{S}_j$ contains s nearest neighbors of Z_j chosen from $\mathbf{Z}_{j-1}$.

4. Compute $\boldsymbol{\sigma}_{j-1}^T(\mathbf{S}_j)\Sigma_{j-1}^{-1}(\mathbf{S}_j)$, then $\mathbf{b}_j(\mathbf{S}_j)$ with 0's in entries corresponding to observations that are not in the set of s nearest neighbors.

5. Find the scalar $V_j(\mathbf{S}_j) = \sigma_j - \boldsymbol{\sigma}_{j-1}^T(\mathbf{S}_j)\Sigma_{j-1}^{-1}(\mathbf{S}_j)\boldsymbol{\sigma}_{j-1}(\mathbf{S}_j)$.

6. $V = \sum_{j=1}^n \mathbf{b}_j(\mathbf{S}_j)V_j^{-1}(\mathbf{S}_j)\mathbf{b}_j^T(\mathbf{S}_j)$ is the sparse approximation to Σ^{-1}.

Notice that $s = n - 1$ gives the exact solution. In practice, choosing the s nearest neighbors to form the conditioning vector is simple and effective. There are other ways to choose conditioning vectors. Jones and Zhang (1997) suggest defining nearest neighbors by the strength of

correlation based on some preliminary estimates, and Stein et al. (2004) argue that including some distant observations in the conditioning vectors shows great gains in efficiency in many circumstances. However, in this paper, we use only nearest neighbors in the conditioning sets to avoid complications, and focus on the size of the conditioning set for the assessment of statistical efficiency.

2.4 Evaluating Statistical Efficiency

Suppose $\boldsymbol{\psi}(\boldsymbol{\theta}) = \mathbf{0}$ is a set of unbiased estimating equations for $\boldsymbol{\theta} \in \mathbb{R}^p$. Let $\dot{\boldsymbol{\psi}}(\boldsymbol{\theta})$ be the $p \times p$ matrix whose jth column is $\partial\boldsymbol{\psi}(\boldsymbol{\theta})/\partial\theta_j$. The efficiency of the estimating equations can be evaluated by the Godambe information matrix

$$\mathbf{G}(\boldsymbol{\theta}) = \big(\mathrm{E}\dot{\boldsymbol{\psi}}(\boldsymbol{\theta})\big)^T \big(\mathrm{E}\boldsymbol{\psi}(\boldsymbol{\theta})\boldsymbol{\psi}(\boldsymbol{\theta})^T\big)^{-1} \big(\mathrm{E}\dot{\boldsymbol{\psi}}(\boldsymbol{\theta})\big),$$

whose inverse gives the approximate covariance matrix of the estimates obtained from these estimating equations (Varin et al., 2011, Stefanski and Boos, 2002). When $\boldsymbol{\psi}(\boldsymbol{\theta}) = \mathbf{0}$ is the exact score equation, $\mathbf{G}(\boldsymbol{\theta}) = \mathbf{I}(\boldsymbol{\theta})$ is the Fisher information matrix.

To calculate the Godambe information matrix, notice that estimating equations (3), (4) and (5) are of the form

$$\psi_i = \mathbf{Z}^T A_i \mathbf{Z} - \mathrm{tr}(A_i\Sigma), \quad i = 1, \dots p,$$

where $A_i = V\Sigma_i\Sigma^{-1}$, $A_i = V\Sigma_i V$ or $A_i = 2V\Sigma_i\Sigma^{-1} - V\Sigma_i V$. When $\Sigma = \phi C(\boldsymbol{\alpha})$, $\boldsymbol{\theta} = (\phi, \boldsymbol{\alpha})$, $\Sigma_1 = C$ and, for $i = 1, \dots, p-1$, $\Sigma_{i+1} = \phi C_i$, where $C_i = \partial C(\boldsymbol{\alpha})/\partial\alpha_i$. When $\Sigma = \phi C(\boldsymbol{\beta}) + \tau I$ with a nugget effect $\tau > 0$, $\boldsymbol{\alpha} = (\boldsymbol{\beta}, \tau)$, $\Sigma_1 = C$, $\Sigma_{i+1} = \phi C_i$ with $C_i = \partial C(\boldsymbol{\beta})/\partial\beta_i$ for $i = 1, \dots, p-2$, and $\Sigma_p = I$. The calculation is straightforward: for $i, j = 1, \dots, p$,

$$\begin{aligned} -E\left(\frac{\partial\psi_i}{\partial\theta_j}\right) &= -\mathrm{tr}\left(\frac{\partial A_i}{\partial\theta_j}\Sigma\right) + \mathrm{tr}\left(\frac{\partial A_i}{\partial\theta_j}\Sigma + A_i\frac{\partial\Sigma}{\partial\theta_j}\right) = \mathrm{tr}(A_i\Sigma_j), \\ \mathrm{cov}(\psi_i, \psi_j) &= 2\,\mathrm{tr}\left(\frac{A_i + A_i^T}{2}\Sigma\frac{A_j + A_j^T}{2}\Sigma\right). \end{aligned}$$

Profile estimating equations (7), (8) and (9) are of the form

$$\psi_1 = \phi - \frac{1}{n}\mathbf{Z}^T C^{-1}\mathbf{Z},$$
$$\psi_{i+1} = \mathbf{Z}^T B_i \mathbf{Z} - \frac{1}{n}\mathbf{Z}^T C^{-1}\mathbf{Z}\,\mathrm{tr}(B_i C), \quad i = 1, \ldots, p-1,$$

where $B_i = VC_iC^{-1}$, $B_i = VC_iV$ or $B_i = 2VC_iC^{-1} - VC_iV$. When $\Sigma = \phi C(\boldsymbol{\alpha})$, $\boldsymbol{\theta} = (\phi, \boldsymbol{\alpha})$, $C_i = \partial C(\boldsymbol{\alpha})/\partial\alpha_i$, $i = 1, \ldots, p-1$. When $\Sigma = \phi\big\{C(\boldsymbol{\beta}) + \delta I\big\}$ with a nugget effect $\tau = \phi\delta > 0$, $\boldsymbol{\alpha} = (\boldsymbol{\beta}, \tau)$, $C^{-1} = \big\{C(\boldsymbol{\beta}) + \delta I\big\}^{-1}$, $C_i = \partial C(\boldsymbol{\beta})/\partial\beta_i$ for $i = 1, \ldots, p-2$, and $C_{p-1} = I$. The approximate variance of the nugget effect $\hat{\tau}$ is given by $\mathbf{u}^T\mathbf{G}^{-1}(\boldsymbol{\theta})\mathbf{u}$, where $\mathbf{u} = (\delta, \mathbf{0}^T, \phi)^T$.

The calculation of the Godambe information matrix is similar. For $i, j = 1, \ldots, p-1$,

$$-E\left(\frac{\partial\psi_1}{\partial\phi}\right) = -1, \quad -E\left(\frac{\partial\psi_1}{\partial\alpha_j}\right) = -\frac{\phi}{n}\mathrm{tr}(C^{-1}C_j),$$
$$-E\left(\frac{\partial\psi_{i+1}}{\partial\phi}\right) = 0, \qquad -E\left(\frac{\partial\psi_{i+1}}{\partial\alpha_j}\right) = \phi\mathrm{tr}\left[\left\{B_i - \frac{1}{n}\mathrm{tr}(B_iC)C^{-1}\right\}C_j\right].$$

Letting $G_i = B_i - \frac{1}{n}\mathrm{tr}(B_iC)C^{-1}$, then

$$\mathrm{var}(\psi_1) = \frac{2\phi^2}{n}, \quad \mathrm{cov}(\psi_1, \psi_{i+1}) = \frac{2\phi^2}{n}\mathrm{tr}\left(\frac{G_i + G_i^T}{2}C\right),$$
$$\mathrm{cov}(\psi_{i+1}, \psi_{j+1}) = 2\phi^2\mathrm{tr}\left(\frac{G_i + G_i^T}{2}C\frac{G_j + G_j^T}{2}C\right).$$

To evaluate the Godambe information matrix, $\mathrm{tr}(VC_iVC)$ and $C^{-1}C_i$ in the trace term require the most computations. The calculation of $\mathrm{tr}(VC_iVC)$ is discussed in the Appendix. Although calculating $C^{-1}C_i$ requires n solves, this calculation only needs to be done once at the estimated parameter values. Therefore, the computation may not be much more difficult than the estimation procedure. When the exact calculation is prohibitive, the stochastic approximation of the trace term proposed by Anitescu et al. (2012) can be considered. As explained in Section 2.1, the computation can be reduced to N solves, where, after preconditioning, Anitescu et al. (2012) found $N \approx 100$ often gives accurate results.

3 Numerical and Simulation Studies

In the numerical and simulation studies, we focus on irregularly spaced data with an unstructured covariance matrix, and a similar simulation design as that in Stein (2013) is used. The observations are taken at the locations $\frac{1}{\sqrt{n}}(r - 0.5 + X_{r\ell}, \ell - 0.5 + Y_{r\ell})$ for $r, \ell \in \{1, 2, \ldots, \sqrt{n}\}$, where the $X_{r\ell}$'s and $Y_{r\ell}$'s are i.i.d. uniform on $(-0.4, 0.4)$ for a total of n observations, and the observations are ordered lexicographically by the ordered pair (r, ℓ).

3.1 Numerical Study

Let $g^{ij}(\boldsymbol{\theta})$ be the elements of $\mathbf{G}^{-1}(\boldsymbol{\theta})$, $i, j = 1, \ldots, p$. In the numerical studies, the efficiency of the estimating equations is evaluated by square roots of the diagonal elements $\sqrt{g^{ii}(\boldsymbol{\theta})}$, $i = 1, \ldots, p$, under two Matérn models: the exponential covariance model $\phi\alpha \exp(-h/\alpha)$; and the Whittle model $2\phi\alpha^2\mathcal{M}_1(h/\alpha)$, where ϕ is the scale parameter, α is the range parameter, and $\mathcal{M}_1 = h\mathcal{K}_1(h)$ with $\mathcal{K}_1$ denoting the modified Bessel function of order 1.

Let $g_e^{ii}(\boldsymbol{\theta})$ denote the diagonal elements of $\mathbf{G}^{-1}(\boldsymbol{\theta})$ for the exact MLE, and $g_a^{ii}(\boldsymbol{\theta})$ be the diagonal elements of $\mathbf{G}^{-1}(\boldsymbol{\theta})$ for the approximate MLEs obtained from any of the three sets of estimating equations (EE) (7)−(9). Figures 1 and 2 give values of $\sqrt{g_a^{ii}(\boldsymbol{\theta})/g_e^{ii}(\boldsymbol{\theta}) - 1}$ (on logarithmic scale), $i = 1, 2$, under the two Matérn models without nugget effect for sample size $n = 900, 2500, 4900$. For each n, we consider conditioning sets consisting of s nearest neighbors when constructing the sparse matrix V with $s = 10, 20, 40, 60, 100$, then compare the statistical efficiency to that of the exact MLEs. For reference, the statistical properties of the exact MLEs are shown in Table 1. When α is known, the exact MLE of ϕ has standard deviation of $\phi\sqrt{2/n}$. The numerical results in Table 1 show that treating α as unknown hardly changes the asymptotic starndard error of ϕ. It is also worth noting that the statistical efficiency for the exact MLE of α hardly changes as n increases, which is expected since α cannot be estimated consistently under fixed domain asymptotics (Zhang, 2004).

Figure 3 gives values of $\sqrt{g_a^{ii}(\boldsymbol{\theta})/g_e^{ii}(\boldsymbol{\theta}) - 1}$ (on logarithmic scale), $i = 1, 2$, under the expo-

Table 1: Properties of (exact) MLEs of $\boldsymbol{\theta} = (\phi, \alpha)$ under exponential covariance functions $\phi\alpha\exp(-h/\alpha)$, and Whittle covariance functions $2\phi\alpha^2\mathcal{M}_1(h/\alpha)$, with n observations in $[0,1]^2$ when $\boldsymbol{\theta} = (1, 0.25)$. Entries are values ($\times 1000$) of $\left(\sqrt{g_e^{11}(\boldsymbol{\theta})}, \sqrt{g_e^{22}(\boldsymbol{\theta})}\right)$ for the MLEs. The last row is the standard deviation ($\times 1000$) of the MLE of ϕ when α is known.

n	900	1600	2500	3600	4900
Exponential	$(48.42, 83.16)$	$(35.98, 81.68)$	$(28.64, 80.82)$	$(23.79, 80.43)$	$(20.35, 79.98)$
Whittle	$(48.54, 56.35)$	$(36.02, 55.29)$	$(28.66, 54.72)$	$(23.80, 54.47)$	$(20.36, 54.16)$
$\sqrt{2/n} \times 1000$	47.14	35.35	28.28	23.57	20.20

nential covariance model with a nugget effect $\tau = 0.001, 0.05, 0.01$ for $s = 10, 20, 40$ and $n = 1600$. The nugget effect, or the discontinuity at the origin in the covariance functions, might reflect microstructure in the underlying process or random measurement error. It is not surprising to see that the nugget effect decreases the statistical efficiency of the estimates overall. The results in Table 2 also show that the presence of a nugget affects ϕ's estimation more than α's. This is also expected, since α does not affect the local behavior of a process, so that a nugget should not have a big influence on $\hat{\alpha}$. The results (not shown) for the Whittle model are quantitively similar. However, this influence on $\hat{\phi}$ is bigger for Whittle covariance models than for exponential models, due to the fact that the Whittle model corresponds to a smoother process than the exponential model, so that the addition of noise makes it harder to estimate its properties.

The numerical results for estimating equations (3)−(5) are nearly identical to those shown in Figures 1−3. For all the cases shown here, when the length of conditioning sets s is 60, the statistical efficiencies of all three approximate MLEs are fairly close to that of the exact MLEs.

Table 2: Properties of estimates of $\boldsymbol{\theta} = (\phi, \alpha, \tau)$ under exponential functions with a nugget effect $\phi\{\alpha\exp(-h/\alpha) + \delta I(h = 0)\}$ for $n = 1600$ observations in $[0,1]^2$ when $(\phi, \alpha) = (1, 0.25)$ and $\tau = \phi\delta = 0.001, 0.005, 0.01$. Entries are values ($\times 1000$) of $\left(\sqrt{g_e^{11}(\boldsymbol{\theta})}, \sqrt{g_e^{22}(\boldsymbol{\theta})}, \sqrt{g_e^{33}(\boldsymbol{\theta})}\right)$ for the exact MLEs.

τ	$\hat{\phi}$	$\hat{\alpha}$	$\hat{\tau}$
0.001	71.07	84.56	1.036
0.005	78.60	85.37	1.277
0.01	85.97	86.19	1.548

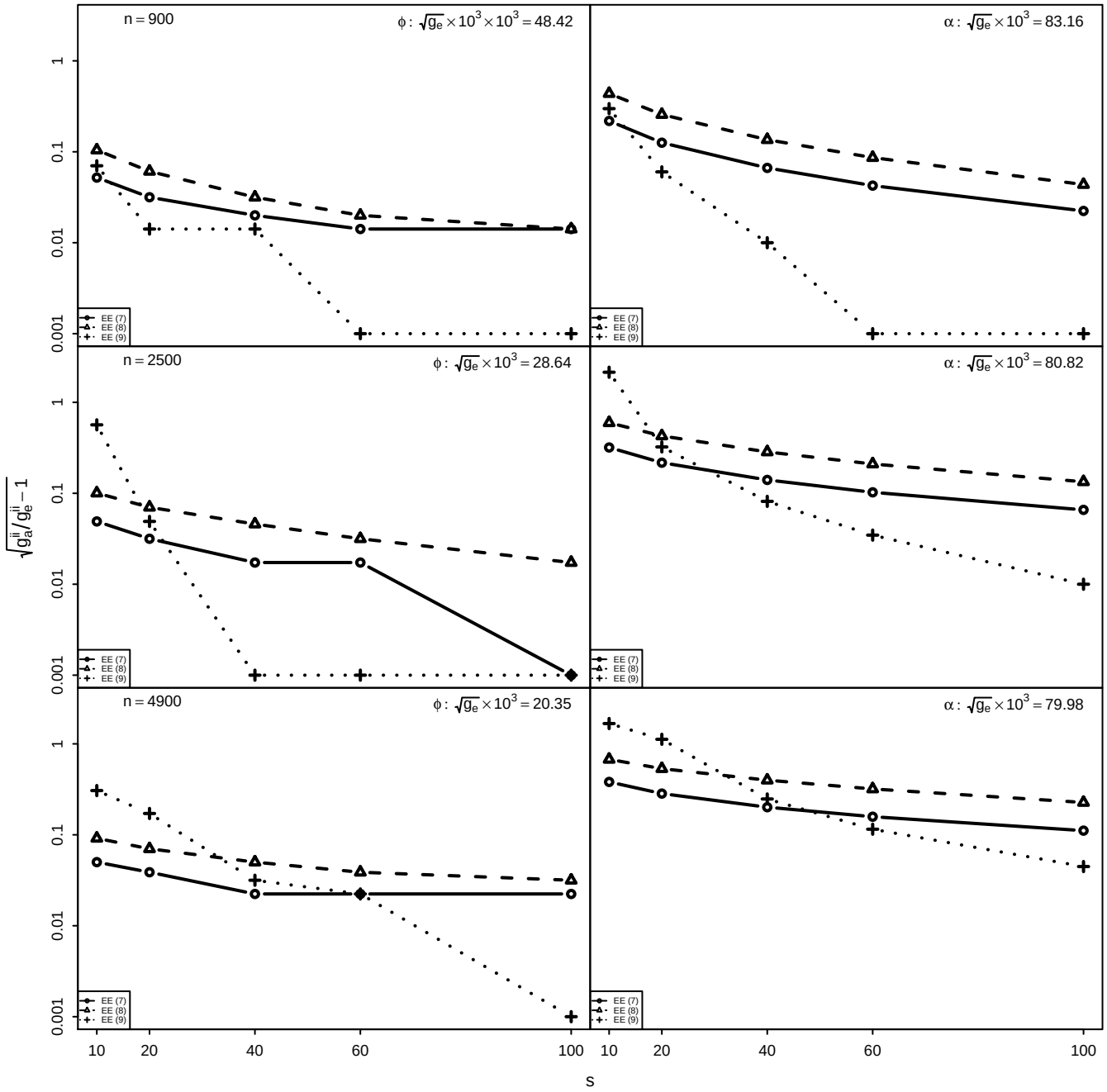

Figure 1: Values of $\sqrt{g_a^{ii}(\boldsymbol{\theta})/g_e^{ii}(\boldsymbol{\theta}) - 1}$ (on logarithmic scale), $i = 1, 2$, for the exact and the approximate MLEs of $\boldsymbol{\theta} = (\phi, \alpha)$ obtained from estimating equations (7)−(9), under the exponential model $\phi\alpha \exp(-h/\alpha)$ with n observations in $[0, 1]^2$ when $\boldsymbol{\theta} = (1, 0.25)$, for $s = 10, 20, 40, 60, 100$ and $n = 900, 2500, 4900$. Smaller values indicate better efficiency. For easy visualization, values less than 0.001 are plotted at 0.001.

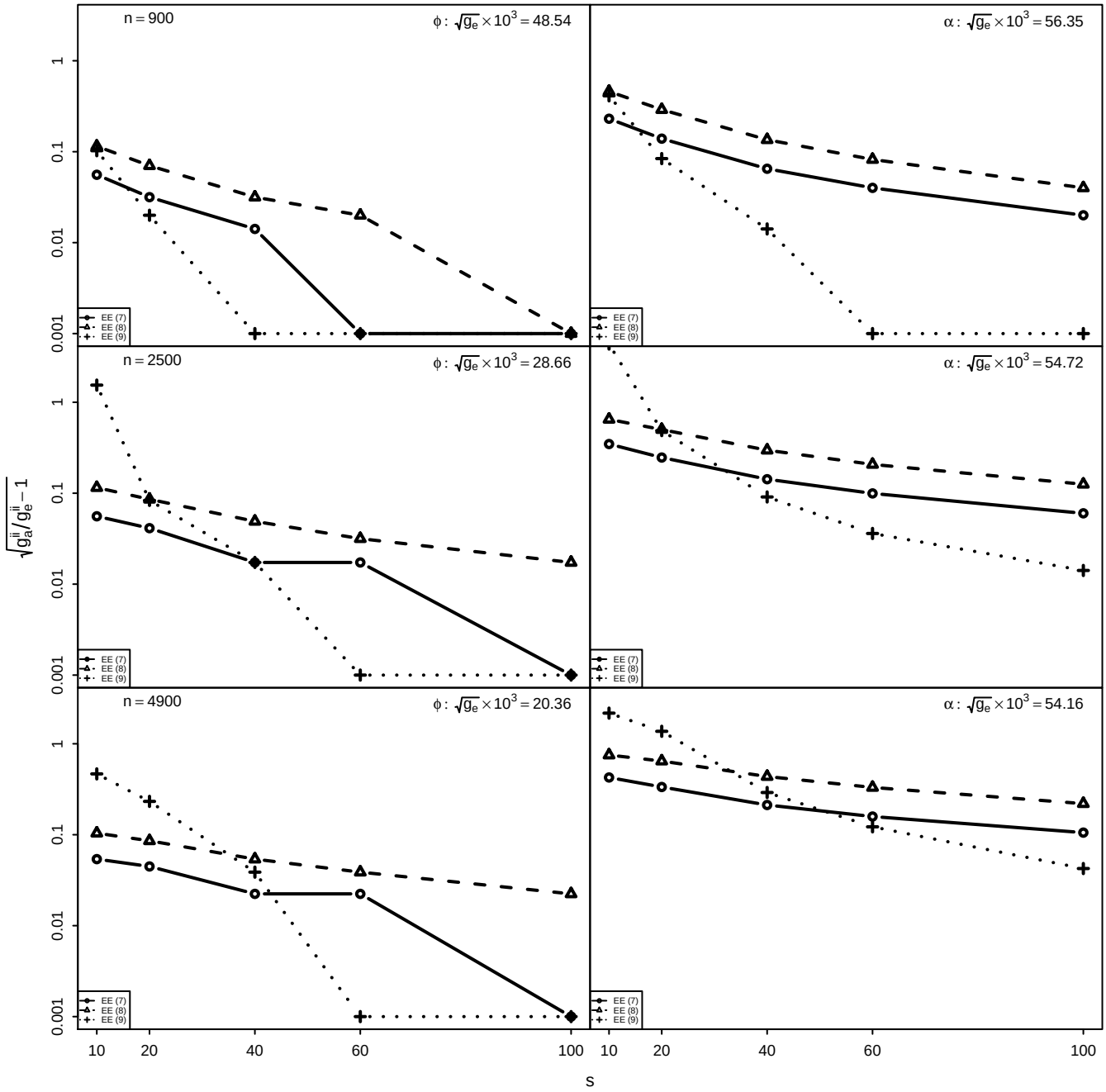

Figure 2: Values of $\sqrt{g_a^{ii}(\boldsymbol{\theta})/g_e^{ii}(\boldsymbol{\theta}) - 1}$ (on logarithmic scale), $i = 1, 2$, for the Whittle model $2\phi\alpha^2\mathcal{M}_1(h/\alpha)$. See Figure 1 for detailed explanations.

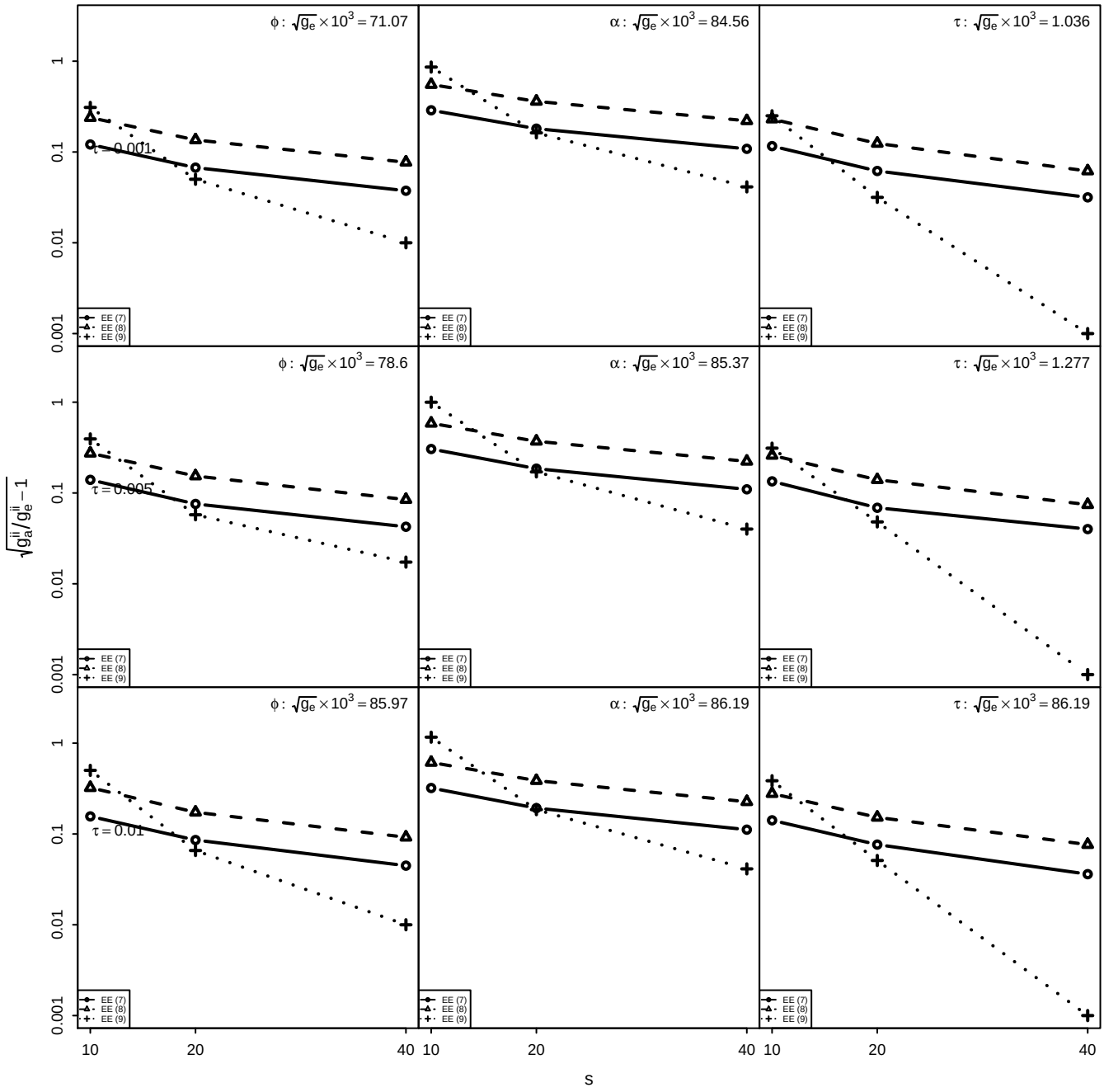

Figure 3: Values of $\sqrt{g_a^{ii}(\boldsymbol{\theta})/g_e^{ii}(\boldsymbol{\theta}) - 1}$ (on logarithmic scale), $i = 1, 2, 3$, for the exponential model with a nugget effect $\phi\{\alpha \exp(-h/\alpha) + \delta I(h = 0)\}$. Settings are similar to Figure 1, but for $n = 1600$, $\tau = \phi\delta = 0.001, 0.005, 0.01$ and $s = 10, 20, 40$.

For $s \geq 60$, the relative efficiency of the approximate MLEs depends only weakly on n. For very small s, the resulting sparse V usually does not approximate C^{-1} well, hence the estimating equations (9) do not necessarily have better statistical efficiency. However, the numerical results clearly show that the estimating equations (9) outperform (7) or (8) for sufficiently large s. Moreover, estimating equations (7) are always better than (8) because VC_iC^{-1} in (7) is a better approximation of $C^{-1}C_iC^{-1}$ than VC_iV in (8).

In principle, the sparse matrix V can be constructed by different approaches. For example, one can divide data into different blocks and assume independence across blocks. This assumption leads to a sparse C^{-1}, which can be used as the matrix V in the estimating equations. Similar to sparse inverse Cholesky, where the length of conditioning sets, s, controls the sparseness of V, the sparseness of V here depends on the block sizes; a small number of blocks, or more data within each block, corresponds to a denser V. We tried this approach and compared the statistical efficiency to the results shown in Figures 1−3, making the sparseness of V comparable to that in the sparse inverse Cholesky approach for each value of s. When V is chosen by the independent blocks method, the statistical efficiencies of all three approximate MLEs obtained from estimating equations (7)−(9) are much worse than those with V constructed by sparse inverse Cholesky, even with a much denser V (results not shown). This suggests that the sparse inverse Cholesky approach is better than independent blocks for choosing V. As the numerical studies in Stein (2013) showed that independent blocks are usually better than isotropic tapers, sometimes by a large margin, the method proposed in this paper is likely much better than the estimating equations with tapered covariance matrices in Kaufman et al. (2008) and Stein (2013), in at least the settings considered in Stein (2013).

The range parameter determines how strong the spatial dependence is. To see if larger α might require larger s to get a given level of statistical efficiency, we compare the statistical efficiency of the range parameter estimates when $\alpha = 0.1, 0.25, 0.5$. Figure 4 shows values of $\sqrt{g_a^{22}(\boldsymbol{\theta})/g_e^{22}(\boldsymbol{\theta}) - 1}$ (on logarithmic scale) for the exact and the approximate MLEs of α obtained

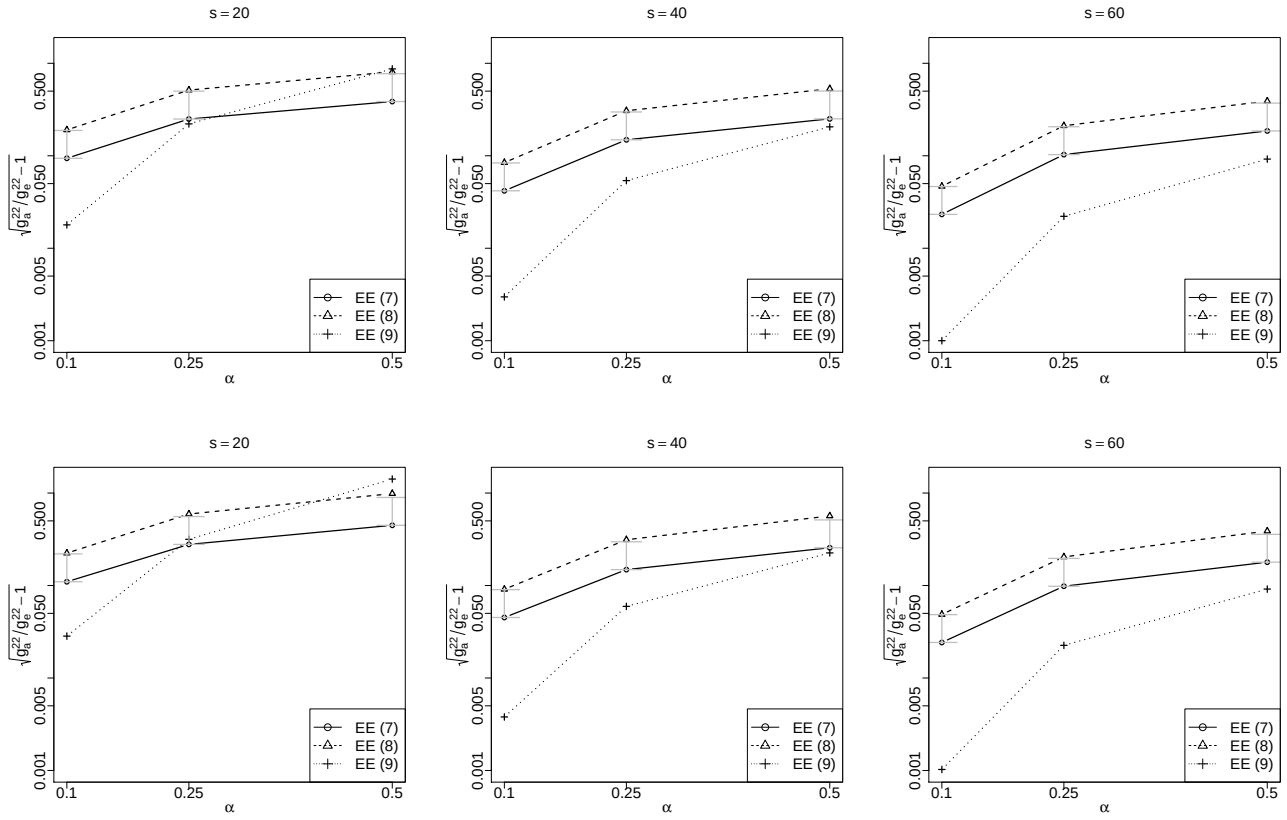

Figure 4: Values of $\sqrt{g_a^{22}(\boldsymbol{\theta})/g_e^{22}(\boldsymbol{\theta}) - 1}$ (on logarithmic scale) for the exact and the approximate MLEs of α obtained from estimating equations (7)−(9), under the exponential model (top panels) and the Whittle model (bottom panels) without nugget effect, for $s = 20, 40, 60$ and $n = 1600$. The vertical bars on each figure indicate values of $\sqrt{g_a^{22}(\boldsymbol{\theta})/g_e^{22}(\boldsymbol{\theta}) - 1}$ that are twice as big as those for estimating equations (7) at $\alpha = 0.1, 0.25, 0.5$. See Figure 1 for plotting details.

from estimating equations (7)−(9), under the exponential and the Whittle models without nugget effect, for $s = 20, 40, 60$ and $n = 1600$. It is clear that $\sqrt{g_a^{22}(\boldsymbol{\theta})/g_e^{22}(\boldsymbol{\theta}) - 1}$ is always larger for (8) than for (7), and by roughly a factor of 2 in all instances. It is not an accident that (7) has only one half the increase in uncertainty over the exact MLE as (8), due to the fact that only one C^{-1} of the quadratic form is replaced by V in (7), while both of them are replaced in (8). Furthermore, as we expect, (9) is much better than (7) and (8) when V is a good approximation to C^{-1}, but can do worse even than (8) when the approximations are not so good. Finally, all estimates do worse relative to the exact MLE as the range parameter α increases, particularly so for estimating equations (9), although (9) is the best for all the models considered when $s \geq 40$.

3.2 Simulation Study

An alternative way to assess the performance of different estimating equations is to repeat the estimation procedure in simulations. For simplicity, we only conduct a simulation using estimating equations (9) with 500 replications for $n = 900$ observations taken from a 30×30 grid on $[0, 1]^2$ with random perturbation under the exponential covariance function with $\phi = 1$ and $\alpha = 0.25$. We consider the sparse matrix V constructed by $s = 10, 20, 40$ nearest neighbors, and for each case we examine the effect of the preconditioner, W, with two different degrees of sparsity, in solving the preconditioned linear system $WC\mathbf{x} = W\mathbf{Z}$ by the conjugate gradient method. Let W_{40} be the sparse matrix V corresponding to $s = 40$, and W_{10} be the sparser matrix corresponding to $s = 10$. Since the variabilities of $\hat{\phi}$ are all close to that of the exact MLE, in Figure 5, only the boxplots of $\hat{\alpha}$ for each s value when using W_{40} are shown. The choice of the preconditioner does not affect the results much. For example, for $s = 40$, the empirical

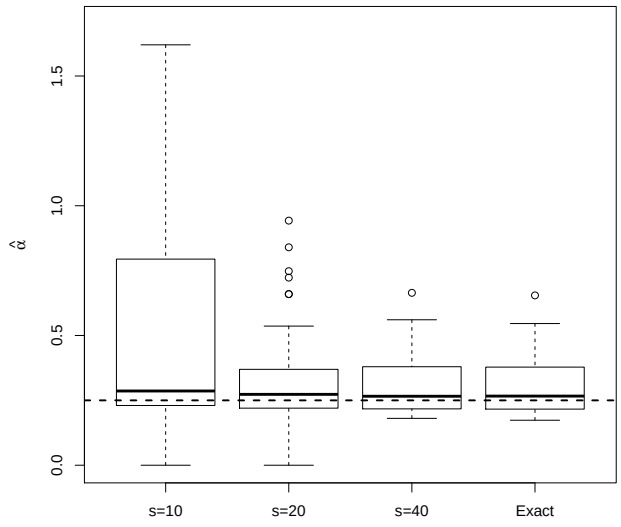

Figure 5: Boxplots of the range parameter estimates for $s = 10, 20, 40$ and the exact MLE with 500 replications when using W_{40}.

variance of the differences in $\hat{\alpha}$ using W_{40} and W_{10} compared to the empirical variance using W_{40} is less than 0.05%, and these ratios are all small for each s value we have considered. However, the preconditioner affects the number of iterations needed to solve the linear system. On average, the iterative method requires 7 iterations for W_{40}, but 13 iterations for W_{10}. Since how quickly iterative methods converge depends on how good the preconditioner is, as expected, to get a given level of accuracy, fewer iterations are required for larger s. Although W_{10} requires more iterations, fewer computations are needed per iteration. For this simulation study, the computation when using W_{10} is faster than using W_{40}.

4 Application to Water Vapor Data

Water vapor is water in its gaseous state and is totally invisible. It is not only a key component of the hydrologic cycle, but also plays vital roles in the chemical reactions on and in atmospheric aerosols. Critically, water vapor is the earth's most important greenhouse gas and is key to understanding the earth's energy budget. To explore the spatial variability of water vapor, we use the Level 2 data product collected by the Moderate Resolution Imaging Spectroradiometer (MODIS) on the National Aeronautics and Space Administration (NASA) satellite from the Aqua platform. The MODIS precipitable water product consists of water vapor amounts as the depth of total precipitable water in centimeter (cm) in the column of air between the surface and the top of the atmosphere at 1 km $\times$ 1 km spatial resolution during daytime. More information can be found at `http://modis-atmos.gsfc.nasa.gov/MOD05_L2/index.html`. One region of the Southeast Pacific Ocean is chosen for this study, where n = 89,479 measurements are observed at UTC 21:10 on January 1, 2009, on a grid of size 300 km $\times$ 300 km with 521 missing values. Figure 6 shows the satellite measurements at 5 km $\times$ 5 km resolution. Since the study region covers a small area near the equator, we assume the area is flat and the grid is equally spaced. The spacing is chosen to be 0.0091 degree, about 1.01 km, which is the distance between two neighboring observations in the center of the region.

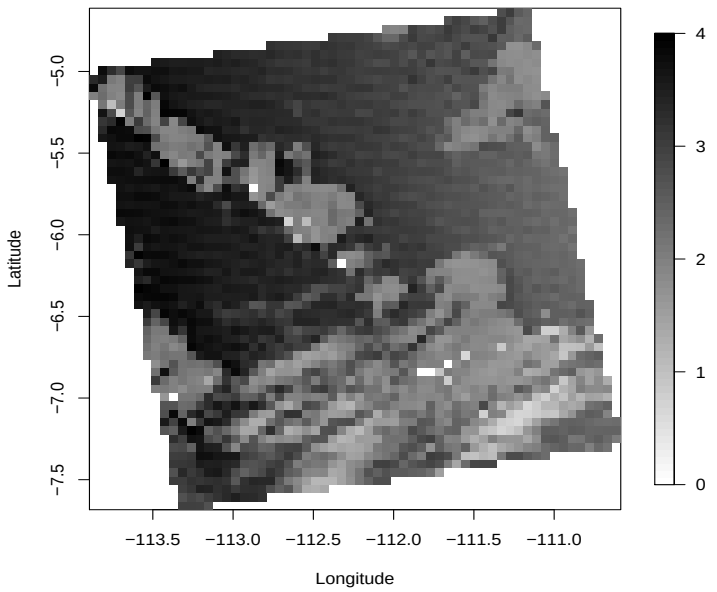

Figure 6: The water vapor measurements (in centimeter) in the Southeast Pacific Ocean region at UTC 21:10 on January 1, 2009 at 5 km $\times$ 5 km resolution with white indicating missing values. Actual resolution is 1 km $\times$ 1km, but plot is at lower resolution to make individual measurements visible.

Figure 7 gives two views of the empirical spatial variogram in various directions. The plots, especially the lower one, indicate that, up to lags of at least 5 km, the variogram is quite close to linear without a nugget effect in all directions and show evidence of an at most modest anisotropy. Therefore, we fit a stationary isotropic Gaussian process model $GP(\mu, K(h, \boldsymbol{\theta}))$ with mean μ and covariance function $K(h) = \phi\alpha \exp(-h/\alpha)$.

Since there is a mean parameter, we apply the REML approach and choose $n-1$ sets of contrasts of $\mathbf{Z} = (Z_1, \ldots, Z_n)^T$ in the parameter estimation procedure, where observations are ordered by rows in a zigzag fashion. Let $\mathbf{Y} = F\mathbf{Z} \sim N(\mathbf{0}, F\Sigma F^T)$, where F is a fixed $(n-1) \times n$ matrix. Here, we take F to be $f_{j,j} = 1$, $f_{j,j+1} = -1$ for $j = 1, \ldots, n-1$, and 0 otherwise, so that $\mathbf{Y}$ consists of differences between adjacent observations.

Parameters are estimated by the set of estimating equations (9) for $\mathbf{Y}$, and V is constructed as a sparse approximation of $(FCF^T)^{-1}$. However, considering (9) is more computationally

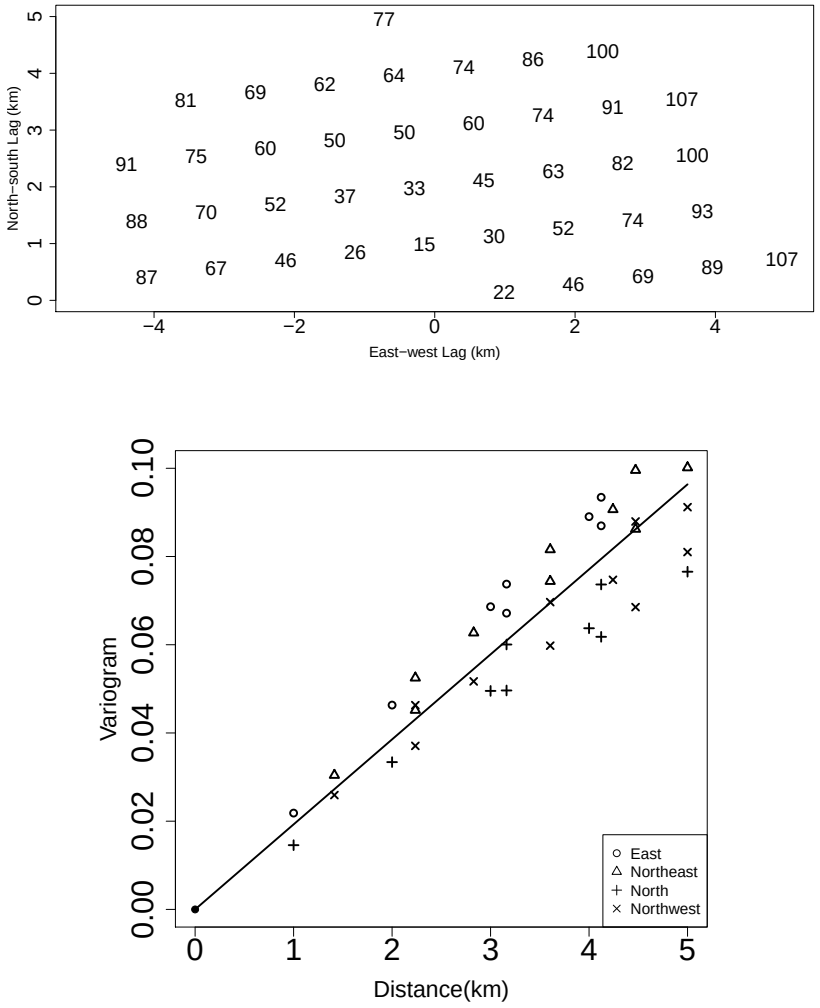

Figure 7: Top panel: the empirical spatial variogram plot. Locations of numbers are lags and the numbers denote variogram values ($\times 1000$). Bottom panel: the directional variogram plot, where different symbols denote different (approximate) directions and the solid line is the isotropic fitting ignoring directions.

demanding but has advantages in statistical efficiency with a good quality of V, we first construct a V by the method described in Section 2.3, where the conditioning sets consist of the contrasts of 60 nearest neighbors, and obtain $\hat{\alpha}_7 = 25.30$ km (shown in Figure 8) and $\hat{\phi}_7 = 0.01743$ by solving (7). Then, we use $(\hat{\phi}_7, \hat{\alpha}_7)$ as initial values in the nonlinear solver, and do a one-step update using (9) with a V that better approximates C^{-1} to obtain an approximate solutions to (9), in which case the conditioning sets contain the contrasts of 112 nearest neighbors. The estimate by one-step update is $(\hat{\phi}_9^{(1)}, \hat{\alpha}_9^{(1)}) = (0.01752, 24.66)$. As shown in Figure 8, one more

updating yields $(\hat{\phi}_9^{(2)}, \hat{\alpha}_9^{(2)}) = (0.01750, 24.48)$, a small change, so we expect that there is not much to be gained by further iterations. The estimated range of the process is $\hat{\alpha} = 24.48$, about 8% of the spatial region, and the estimated variance of the process, $\hat{\phi}\hat{\alpha} = 0.43$. The asymptotic standard errors for $\hat{\phi}$ and $\hat{\alpha}$ at the estimated values are 0.00477 and 21.57, respectively. The standard error of $\hat{\alpha}$ is nearly as large as the estimate itself, suggesting we should not take this standard error too seriously except as an indication that the range parameter is very difficult to estimate using these data.

In this application, the covariance matrix is of size 89,479 × 89,479, which would make computing even a single linear solve with it very difficult without exploiting the structure of the matrix. In particular, by treating the region as flat, the observations then form a regular grid with some missing values. Despite the missing values, we are able to take advantage of such a

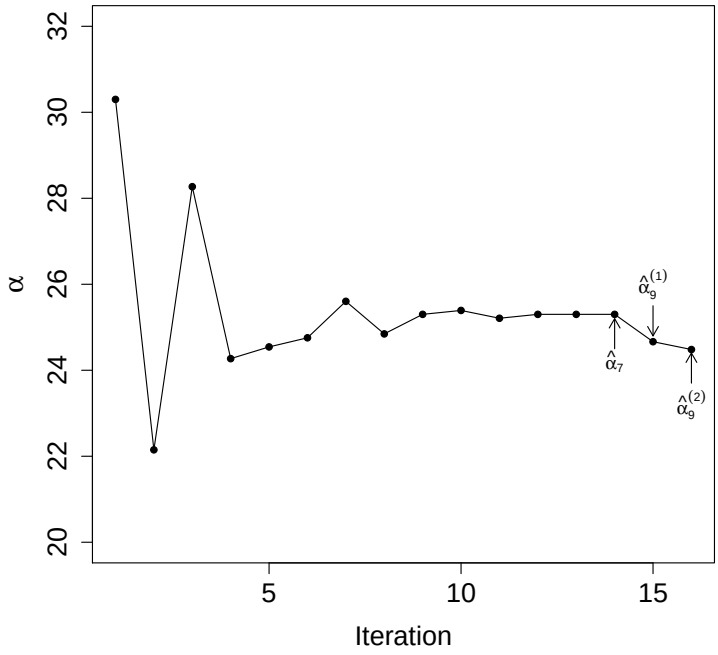

Figure 8: Estimation of α: $\hat{\alpha}_7$ is the estimate by solving estimating equations (7), $\hat{\alpha}_9^{(1)}$ is the estimate using (9) with one-step update, and $\hat{\alpha}_9^{(2)}$ is the estimate by doing one more update using (9).

structure to handle the large matrix-vector multiplication by circulant embedding (Kozintsev, 1999), as well as reduce the computations required for constructing the V matrix. Details are provided in the Appendix.

All the computations reported on here were done by a single threaded application in `Matlab` on a 2.80Ghz Intel Xeon X5560 with 48GB of RAM. It takes only about 100 seconds per α value for the iterative method to solve the linear system, due to the fast matrix-vector multiplication by FFT and the preconditioning. To solve the estimating equation (7) requires around 3,000 seconds per α value to find V with 60 nearest neighbors, and 800 seconds to compute the trace term. For the estimating equation (9), it takes about 9,000 seconds to find V with 112 nearest neighbors, and 13,000 seconds to calculate the trace term. By using parallel computing, we are able to compute the asymptotic standard errors within one day.

5 Discussion

We have proposed three new unbiased estimating equations that are both computationally and statistically efficient for fitting Gaussian process models on $\mathbb{R}^d$ based on large, irregularly spaced sets of observations. There are many other approximation methods for large spatial datasets. Each has their strengths and weaknesses, but not all extend naturally to models on $\mathbb{R}^d$. For example, Markov Random Field models depend on the observation locations, and realignment to a much finer grid with missing values is required for irregular locations (Lindgren et al., 2011). Our proposed methods yield unbiased estimating equations under models that are valid and consistent throughout $\mathbb{R}^d$, and provide a useful combination of statistical and computational efficiency.

This paper focused on approximating the precision matrix Σ^{-1} directly by a sparse matrix and obtained computationally less demanding sets of unbiased estimating equations to avoid multiple solves. For example, the estimating equations in (7) avoid the multiple solves (often around 100) that are needed in Stein et al. (2012) to get accurate approximations to the score

function. The main computation in our approach is finding the sparse approximation of the precision matrix and matrix-vector multiplication in the parameter estimation procedure; some details are provided in the Appendix. Our methods are suitable for irregularly spaced large spatial datasets; however, depending on specific applications, some exploitable structures in the covariance matrix, such as sparseness or embeddability in circulant matrices, can further reduce the computational cost.

In this work, we constructed a sparse V by a sparse inverse Cholesky approach. Since the statistical properties depend on how well the matrix V approximates Σ^{-1}, other computationally efficient methods of finding a good sparse approximation can be further explored. In the iterative method, to precondition Σ, we generally used the same V as in the estimating equations. Since it is not essential that this V be used to precondition, other effective preconditioning methods can be investigated. Indeed, in our simulation study, we found that it may be faster to use a cruder but faster preconditioner than V. Moreover, to reduce computations further, it is also worthwhile to develop useful techniques to achieve faster convergence of the nonlinear solver when solving the estimating equations.

In the numerical studies, we considered exponential and Whittle covariance functions, which are two special cases of the Matérn family (Stein, 1999). The Matérn model is a flexible parametric class with scale parameter ϕ, range parameter α, and smoothness parameter ν that controls the smoothness of a random field. For many applications, allowing ν in the Matérn model to be unknown would be desirable. However, we would then need to evaluate the derivative of the Matérn model with respect to ν. The convergent and asymptotic series for this derivative can be found in Olver, Lozier, Boisvert and Clark (2010, Chapter 10), although adapting these into accurate and efficient code is not a simple task. A finite difference approximation of the derivative with respect to ν might be adequate in some circumstances. For large spatial datasets, one often needs to use nonstationary covariance functions. Our approach can be applied to parametric families of nonstationary covariance functions for which all the derivatives can be computed. For

example, some nonstationarities can be parameterized based on known covariates (e.g., covariance structures that change with latitude or with land type), in which case, derivatives with respect to these parameters might be available.

Acknowledgement

This research was partially supported by the US National Science Foundation grants DMS-1106862, 1106974 and 1107046, the STATMOS research network on Statistical Methods in Oceanic and Atmospheric Sciences. The authors thank the anonymous reviewers for their valuable comments.

References

Anitescu, M., Chen, J. and Wang, L. (2012), "A matrix-free approach for solving the Gaussian process maximum likelihood problem," *SIAM Journal on Scientific Computing*, 34, 1, 240-262.

Aune, E., Simpson, D. and Eidsvik, J. (2013), "Parameter estimation in high dimensional Gaussian distributions," *Journal of Statistics and Computing*, 247-263.

Caragea, P. (2003), "Approximate likelihoods for spatial processes," *Ph.D. Dissertation*, University of North Carolina at Chapel Hill.

Chen, K. (2005), *Matrix Preconditioning Techniques and Applications,* Cambridge University Press, Cambridge, UK.

Fang, K. F., Li, R. Z. and Sudjianto, A. (2005), *Design and Modeling for Computer Experiments*, New York: Chapman & Hall/CRC Press.

Fuentes, M. (2007), "Approximate likelihood for large irregularly spaced spatial data," *Journal of the American Statistical Association*, 102, 321-331.

Furrer, R., Genton, M. G. and Nychka, D. (2006), "Covariance tapering for interpolation of large spatial datasets," *Journal of Computational and Graphical Statistics*, 15, 502-523.

Genton, M. G. (2007), "Separable approximations of space-time covariance matrices," *Environmetrics*, 18, 681-695.

Gneiting, T., Genton, M. G. and Guttorp, P. (2007), "Geostatistical space-time models, stationarity, separability and full symmetry," in *Statistics of Spatio-Temporal Systems*, (eds Finkenstaedt, B., Held, L. and Isham, V.), Chapman & Hall/CRC Press, 151-175.

Hutchinson, M. F. (1990). "A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines," *Communications in Statistics- Simulations*, 19, 433-450.

Jones, R. H. and Zhang, Y. (1997), "Models for continuous stationary space-time processes," in *Modelling Longitudinal and Spatially Correlated Data* (eds T. G. Gregoire, D. R. Brillinger, P. J. Diggle, E. Russek-Cohen, W. G. Warren and R. D. Wolfinger), pp. 289-298. New York: Springer.

Kaufman, C., Schervish, M. and Nychka, D. (2008), "Covariance tapering for likelihood-based estimation in large spatial datasets," *Journal of the American Statistical Association*, 103, 1556-1569.

Kitanidis, P. K. (1983), "Statistical estimation of polynomial generalized covariance functions and hydrologic applications," *Water Resources Research*, 19, 909-921.

Kolotilina, L. Y. and Yeremin, A. Y. (1993), "Factorized sparse approximate inverse preconditioning I. Theory," *SIAM Journal on Matrix Analysis and Applications*, 14, 45-58.

Kozintsev, B. (1999), "Computations With Gaussian Random Fields," PhD Thesis, ISR99-3, University of Maryland.

Lindgren, F., Rue, H. and Lindström, J. (2011), "An explicit link between Gaussian fields and Gaussian Markov random fields: The SPDE approach (with discussion)," *Journal of the Royal Statistical Society Series B*, 73, 423-498.

Olver, F. W. J., Lozier, D. W. , Boisvert, R. F. and Clark, C. W. (2010), *NIST Handbook of Mathematical Functions,* Cambridge University Press, New York, NY.

Rue, H. and Tjelmeland, H. (2002), "Fitting Gaussian Markov random fields to Gaussian fields," *Scandinavian Journal of Statistics*, 29, 31-50.

Santner, T. J., Williams, B. J. and Notz, W. I. (2003), *The Design and Analysis of Computer Experiments*, New York: Springer.

Stefanski, L. A. and Boos, D. D. (2002), "The calculus of M-estimation," *Journal of the American Statistical Association*, 56, 29-38.

Stein, M. L. (1999), *Interpolation of Spatial Data: Some Theory for Kriging,* New York: Springer.

Stein, M. L. (2013), "Statistical properties of covariance tapers," *Journal of Computational and Graphical Statistics*, 22, 866-885.

Stein, M. L., Chen, J. and Anitescu, M. (2012), "Stochastic approximation of score functions for Gaussian processes," *Annals of Applied Statistics*, to appear.

Stein, M. L., Chi, Z. and Welty, L. J. (2004), "Approximating likelihoods for large spatial datasets," *Journal of the Royal Statistical Society Series B*, 66 275-296.

Sun, Y., Li, B., and Genton, M. G. (2012), "Geostatistics for large datasets," in *Advances And Challenges In Space-time Modelling Of Natural Events*, (eds J. M. Montero, E. Porcu, M. Schlather), Springer, Vol. 207, Chapter 3, 55-77.

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

Varin, C., Reid, N. and Firth, D. (2011), "An Overview of Composite Likelihood Methods," *Statistica Sinica*, 21, 542.

Zhang, H. (2004), "Inconsistent estimation and asymptotically equal interpolations in model based Geostatistics, *Journal of the American Statistical Association*, 99, 250-261.

Appendix: Computational Details

Calculations of $\mathrm{tr}(VC_iVC)$

In estimating equations (8) and (9), a natural way to calculate the trace term $\mathrm{tr}(VC_iVC)$ is to exploit the sparseness of V to compute VC_i and VC and then the diagonal elements of their product. When we are not able to store the entire covariance matrix C or its derivative C_i, this calculation requires repeatedly calculating the elements of C_i if we compute one row of VC_i and one column of VC at a time to obtain the diagonal elements of VC_iVC. A better way is to calculate one row of VC_iV and one column of C at a time.

To show how to compute the jth row of VC_iV, let $\mathbf{v}_j^T$ be the jth row of V, $j = 1, \ldots, n$. We first calculate $\mathbf{v}_j^T C_i V$ (some techniques and computational requirements for computing $\mathbf{v}_j^T C_i$ are discussed in Section 2.2), then $\mathbf{v}_j^T C_i V$ can be easily calculated since we are able to store the sparse matrix V.

Construction of V **for data on a grid**

When data are on a grid, for most observations in the interior of the spatial domain, the conditioning sets consisting of s nearest neighbors form the same distance matrix; for example, in Figure 9, when observations are ordered by rows from bottom left in a zigzag fashion, the shaded 12 nearest neighbors of the observations A, B and C have the same distance matrix under the isotropic assumption, thus the same $V_j(\mathbf{S}_j)$ defined in Section 2.3 can be used repeatedly. The conditional distribution of an observation requires new computations only when it is near an edge or missing values are present in its neighbors. This structure can be used to save further computational time.

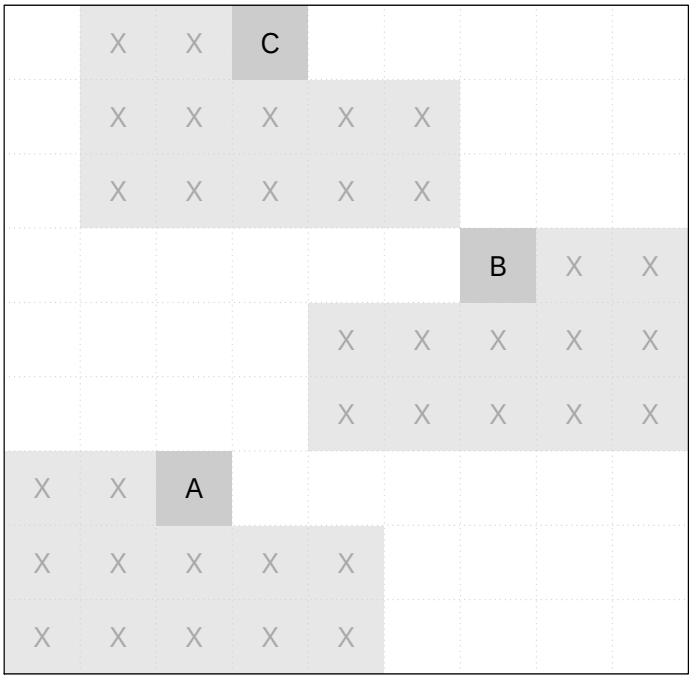

Figure 9: The 12 nearest neighbors of the observations A, B and C, when observations are ordered by rows from bottom left in a zigzag fashion.