1-16-2013

Estimation and Prediction in Spatial Models With Block Composite Likelihoods

Jo Eidsvik
University of Trondeim

Benjamin A. Shaby
University of California - Berkeley

Brian J. Reich
North Carolina State University at Raleigh

Matthew Wheeler
University of California, Santa Barbara

Jarad Niemi
Iowa State University, niemi@iastate.edu

Estimation and prediction in spatial models with block composite likelihoods

Jo Eidsvik[1], **Benjamin A. Shaby**[2], **Brian J. Reich**[3], **Matthew Wheeler**[4] **and Jarad Niemi**[5]

1) Department of Mathematical Sciences, NTNU, 7491 Trondheim, Norway.

(joeid@math.ntnu.no)

2) Department of Statistics, University of California, Berkeley, Berkeley, California 94720, U.S.A.

(bshaby@stat.berkeley.edu)

3) Department of Statistics, North Carolina State University, Raleigh, North Carolina 27695,

U.S.A. (brian_reich@ncsu.no)

4) Department of Statistics and Applied Probability, University of California at Santa Barbara,

Santa Barbara, California 93106, U.S.A. (wheeler@pstat.ucsb.edu)

5) Department of Statistics, Iowa State University, Ames, IA 50011, U.S.A. (niemi@iastate.edu)

Abstract

A block composite likelihood is developed for estimation and prediction in large spatial datasets. The composite likelihood is constructed from the joint densities of pairs of adjacent spatial blocks. This allows large datasets to be split into many smaller datasets, each of which can be evaluated separately, and combined through a simple summation. Estimates for unknown parameters are obtained by maximizing the block composite likelihood function. In addition, a new method for optimal spatial prediction under the block composite likelihood is presented. Asymptotic variances for both parameter estimates and predictions are computed using Godambe sandwich matrices. The approach gives considerable improvements in computational efficiency, and the composite structure obviates the need to load entire datasets into memory at once, completely avoiding memory limitations imposed by massive datasets. Moreover, computing time can be reduced even further by distributing the operations using parallel computing. A simulation study shows that composite likelihood estimates and predictions, as well as their corresponding asymptotic confidence intervals, are competitive with those based on the full likelihood. The procedure is demonstrated on one dataset from the mining industry and one dataset of satellite retrievals. The real-data examples show that the block composite results tend to outperform two competitors; the predictive process model and fixed rank Kriging. Supplemental material for this article is available online.

Keywords: large datasets, spatial statistics, parallel computing, GPU, Gaussian process

1 Introduction

In recent years there has been a tremendous increase in the magnitude and pervasiveness of massive geocoded scientific datasets. The growth in size is to a large extent driven by new technologies such as GPS and remote sensing, as well as by the ever-increasing storage capacity of digital databases. The explosion of interest in climate research has brought these types of datasets into the spotlight. These developments have triggered demand for more sophisticated statistical modeling and methodology for such data. The computations required for inference and prediction in spatial Gaussian process models, the central construct in spatial statistics, are challenging for large datasets because they require manipulations of large covariance matrices. In particular, evaluation of the likelihood function necessitates performing inverse and determinant operations, both of which are computationally intractable for large matrices.

Several approaches have been proposed to mitigate this computational burden. Furrer et al. (2006), Kaufman et al. (2008), and Shaby and Ruppert (2012) use covariance tapering, where the full covariance function is multiplied by a compactly-supported correlation function, yielding a sparse covariance matrix, which enables specialized algorithms to be leveraged. Another strategy is to represent the spatial process in a lower-dimensional subspace using low-rank models (e.g. Stein, 2008; Cressie and Johannesson, 2008; Banerjee et al., 2008). Gaussian Markov random fields are also useful for fast computation (Lindgren et al., 2011; Aune et al., 2012).

In this paper we implement a unified framework for parameter estimation and prediction based on the composite likelihood (CL) (Lindsay, 1988; Varin, 2008; Varin et al., 2011). The CL is a product of several joint likelihoods of subsets of the data. One important special case is the pairwise likelihood, which is the product of all bivariate marginal likelihoods. Here, we use a form of the CL function defined as the product of joint density functions of pairs of spatial blocks. The motivation behind the spatial blocking strategy is that it captures much of the spatial dependence, while still providing the divide and conquer aspect of the CL, which reduces computational complexity and facilitates fast parallel computing. Unlike low-rank basis methods (Stein, 2008; Cressie and Johannesson, 2008; Banerjee et al., 2008) and embedded lattice methods (Lindgren

et al., 2011), estimation and prediction with block CLs allows one to work directly with full-rank continuous spatial processes.

In the parameter estimation context, the asymptotic properties of the CL are well-understood. Maximum CL estimates are consistent and asymptotically normal (Varin, 2008) under similar conditions as corresponding maximum likelihood estimates (i.e. under expanding domain asymptotics (Mardia and Marshall, 1984)). The asymptotic covariance for maximum CL estimates is given by a sandwich matrix (Godambe, 1960; Godambe and Heyde, 1987) rather than the usual Fisher information matrix for maximum likelihood estimators.

In addition to parameter estimation for Gaussian random fields, we show how to use the CL for the crucial complementary problem of spatial prediction, which has not previously been considered. We demonstrate how to construct predictions at unobserved sites that are optimal under the block CL, the composite analogue to Kriging. This approach allows fast spatial prediction. We derive asymptotic prediction variances under the CL model, which have the familiar sandwich form.

The earliest use of the CL for random fields seems to be Heagerty and Lele (1998) and Curriero and Lele (1999), who used the pairwise form of the CL to estimate covariance parameters, and establish consistency and asymptotic normality. Several attempts have been made to utilize spatial blocking. Among them is Caragea and Smith (2007), who use averages of big blocks or products of small block likelihoods, to construct estimators. They also use hybrid schemes combining their big and small blocks methods, and study their asymptotic properties. Their ways of combining blocks are different from our block CL approach. Stein et al. (2004) use a restricted likelihood version of the telescoping conditional probability approximation of Vecchia (1988), which achieves fast computations by reducing the conditioning set to a small subset of the data. It is not obvious how to consistently combine parameter estimation and spatial prediction with this approach.

Whereas evaluating the likelihood requires $O(n^3)$ operations, the block CL model reduces the computational burden to $O(n)$, where the hidden constant would depend on the block sizes. Moreover, the usual memory restrictions for large datasets are avoided since the blocks of data can be loaded into memory separately. Finally, the CL approach allows parallel computing. We test the

estimation procedure on a Graphical Processing Unit (GPU) and achieve a many-fold increase in computational efficiency. This speed-up comes on top of the efficiency gains resulting from the structure of the CL function. Graphics cards have evolved into massively-parallel computational engines. Statisticians are beginning to exploit the technology (Suchard and Rambaut, 2009; Suchard et al., 2010; Lee et al., 2010).

Section 2 defines the block CL function. Methods for estimation and prediction are presented in Section 3, and computational methods and parallelization issues are described in Section 4. Section 5 provides simulation studies. Section 6 presents two data examples. The computational details related to the CL function are in the Appendix (Supplementary material).

2 Block composite likelihood for spatial data

2.1 Geostatistical model

Geostatistical settings typically assume, at locations $\boldsymbol{s} \in D \subseteq \Re^d$, $d = 2$ or 3, a Gaussian response variable $Y(\boldsymbol{s})$ along with a $p \times 1$ vector of spatially-referenced explanatory variables $\boldsymbol{x}(\boldsymbol{s})$ which are associated through a spatial regression model

$$Y(\boldsymbol{s}) = \boldsymbol{x}^t(\boldsymbol{s})\boldsymbol{\beta} + w(\boldsymbol{s}) + \epsilon(s), \tag{1}$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^t$ is the regression parameter, and $\epsilon(s) \sim N(0, \tau^2)$ is independent error. The spatial residual $w(\boldsymbol{s})$ provides structural dependence, capturing the effect of unobserved covariates with spatial pattern. The covariance structure of the Gaussian process $w(\boldsymbol{s})$ is typically characterized by a small number of parameters. We denote the collection of all covariance parameters $\boldsymbol{\theta}$, which includes the nugget effect τ^2. Some common models for $\text{Cov}(w(\boldsymbol{s}), w(\boldsymbol{s}^{'})) = C(\boldsymbol{s}^{'}, \boldsymbol{s})$ are the exponential, Matérn and Cauchy covariance models.

We assume data are available at n locations $\{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_n\}$, and denote the collection of data $\boldsymbol{Y} = (Y(\boldsymbol{s}_1), \ldots, Y(\boldsymbol{s}_n))^t$. Then $\boldsymbol{Y} \sim N(\boldsymbol{X}\boldsymbol{\beta}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\boldsymbol{\theta}) = \boldsymbol{C} + \tau^2\boldsymbol{I}_n$, with $C(i,j) = \text{Cov}(w(\boldsymbol{s}_i), w(\boldsymbol{s}_j))$. Moreover, row i of matrix $\boldsymbol{X}$ contains the explanatory variables

$\boldsymbol{x}^t(\boldsymbol{s}_i)$. Ignoring a scalar that does not depend on $\boldsymbol{\beta}$ or $\boldsymbol{\theta}$, the log likelihood is

$$\ell(\boldsymbol{Y};\boldsymbol{\beta},\boldsymbol{\theta}) = -\frac{1}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})^t\boldsymbol{\Sigma}^{-1}(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta}). \tag{2}$$

Noting that $\boldsymbol{\Sigma}$ is $n \times n$, the difficulty with the usual maximum likelihood method is apparent; evaluating the log-likelihood requires computing $|\boldsymbol{\Sigma}|$ and a quadratic form that includes $\boldsymbol{\Sigma}^{-1}$, both of which are computationally intractable for large n.

2.2 Composite likelihood

In contrast to the full joint likelihood (2), the CL function (Lindsay, 1988) is constructed from marginal likelihoods of subsets of the data, proceeding as though these subsets were independent. Curriero and Lele (1999) use a spatial CL model based on pairwise data differences.

We next present a block CL, where we partition the region D into M blocks $D_1, \ldots, D_M$, with $\cup_k D_k = D$, $D_k \cap D_l = \emptyset$, for all pairs of blocks k, l. Denote the response in block $k = 1, \ldots, M$ as $\boldsymbol{Y}_k = \{Y(\boldsymbol{s}_i); \boldsymbol{s}_i \in D_k\}$. The number of sites in block k is n_k, $\sum_k n_k = n$. Let $\boldsymbol{Y}_{kl} = (\boldsymbol{Y}_k^t, \boldsymbol{Y}_l^t)^t$ be the collection of data in blocks k and l. We define the block composite log likelihood as

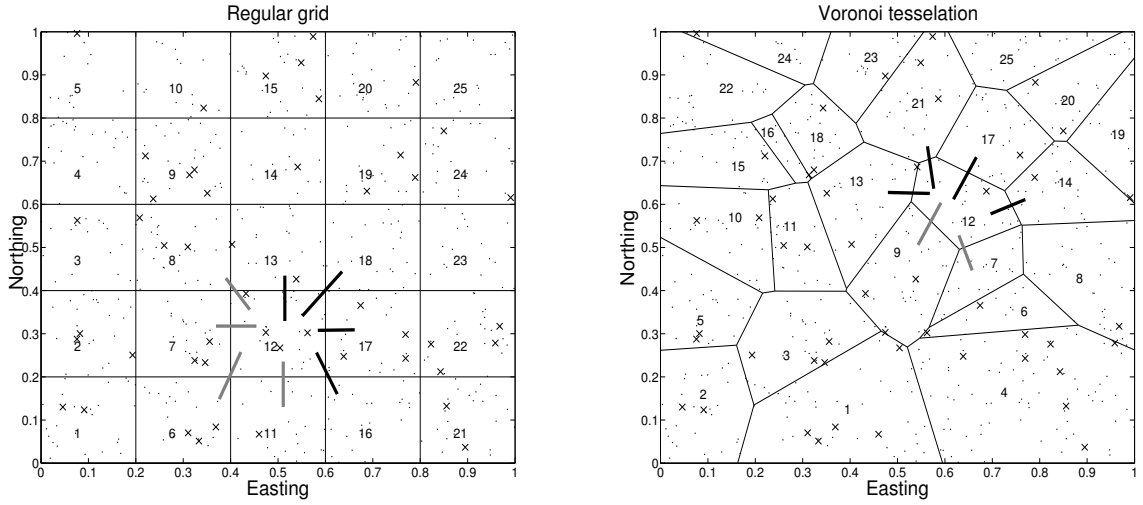
$$\begin{aligned}\ell_{CL}(\boldsymbol{Y};\boldsymbol{\beta},\boldsymbol{\theta}) &= \sum_{k=1}^{M-1}\sum_{l>k}\ell(\boldsymbol{Y}_{kl};\boldsymbol{\beta},\boldsymbol{\theta}) \\ &= \sum_{k=1}^{M-1}\sum_{l>k}\Big[-\frac{1}{2}\log|\boldsymbol{\Sigma}_{kl}| - \frac{1}{2}(\boldsymbol{Y}_{kl}-\boldsymbol{X}_{kl}\boldsymbol{\beta})^t\boldsymbol{\Sigma}_{kl}^{-1}(\boldsymbol{Y}_{kl}-\boldsymbol{X}_{kl}\boldsymbol{\beta})\Big].\end{aligned} \tag{3}$$

Here, $\boldsymbol{X}_{kl} = (\boldsymbol{X}_k^t, \boldsymbol{X}_l^t)^t$ is the collection of all covariates in block k and l, and $\boldsymbol{\Sigma}_{kl}$ is the $(n_k + n_l) \times (n_k + n_l)$ covariance matrix

$$\boldsymbol{\Sigma}_{kl} = \begin{bmatrix} \boldsymbol{\Sigma}_{kl}(1,1) & \boldsymbol{\Sigma}_{kl}(1,2) \\ \boldsymbol{\Sigma}_{kl}(2,1) & \boldsymbol{\Sigma}_{kl}(2,2) \end{bmatrix}, \tag{4}$$

where $\boldsymbol{\Sigma}_{kl}(1,1)$ is the $n_k \times n_k$ covariance matrix of $\boldsymbol{Y}_k$, $\boldsymbol{\Sigma}_{kl}(2,2)$ is the $n_l \times n_l$ covariance matrix of $\boldsymbol{Y}_l$, and $\boldsymbol{\Sigma}_{kl}(1,2) = \boldsymbol{\Sigma}_{kl}^t(2,1)$ is the $n_k \times n_l$ cross-covariance between $\boldsymbol{Y}_k$ and $\boldsymbol{Y}_l$. If $M = 1$ or 2, the block CL in (3) is equal to the full likelihood in (2); if $M = n$, we get the pairwise likelihood. The block CL is a natural compromise for spatial models, as the number of blocks M represents a trade-off between computational and statistical efficiency.

The CL expression is simplified by omitting distant pairs of blocks from expression (3), assuming negligible dependence between blocks if they are not neighbors. Let N_k denote the neighbors of block k. Figure 1(a) shows an illustration of a regular block design with $M = 5 \cdot 5 = 25$ blocks on a $2D$ domain, while Figure 1(b) shows a Voronoi / Delaunay design. Here, blocks are neighbors if they share a common border. A neighbor structure is easy to represent as a graph. The edges of block 12 are shown in Figure 1. In Figure 1(b) block 7 has a border with 14, and this prevents block 8 from being a neighbor of 12.



(a) (b)

Figure 1: Observation sites illustrated by '.' and predictions sites by 'x'. A block CL splits the spatial domain into regular (a) or irregular (b) blocks. Each block communicates pairwise with each of its neighbors. For the regular grid (a), an interior block has eight neighbors. For a random or adaptive grid (b), the number of neighbors varies. In both displays the block indexed 12 has four neighbors with higher indices (black edges).

This neighbor set N_k can be split into a forward part $N_k^{\rightarrow} = \{l > k\} \cap \{l \in N_k\}$ and a backward part $N_k^{\leftarrow} = \{l < k\} \cap \{l \in N_k\}$. These two are displayed using black and gray edge lines in Figure 1. By only considering these neighboring blocks, the second sum in (3) is only

evaluated over $l \in N_k^{\rightarrow}$, so that

$$\ell_{CL}(\boldsymbol{Y};\boldsymbol{\beta},\boldsymbol{\theta}) = \sum_{k=1}^{M-1}\sum_{l\in N_k^{\rightarrow}}\Big[-\frac{1}{2}\log|\boldsymbol{\Sigma}_{kl}| - \frac{1}{2}(\boldsymbol{Y}_{kl}-\boldsymbol{X}_{kl}\boldsymbol{\beta})^t\boldsymbol{\Sigma}_{kl}^{-1}(\boldsymbol{Y}_{kl}-\boldsymbol{X}_{kl}\boldsymbol{\beta})\Big] \tag{5}$$
$$= \sum_{j}\Big[-\frac{1}{2}\log|\boldsymbol{\Sigma}_{j}| - \frac{1}{2}(\boldsymbol{Y}_{j}-\boldsymbol{X}_{j}\boldsymbol{\beta})^t\boldsymbol{\Sigma}_{j}^{-1}(\boldsymbol{Y}_{j}-\boldsymbol{X}_{j}\boldsymbol{\beta})\Big].$$

The shorthand notation with index j represents the set of edges $(k,l) \mid l \in N_k^{\rightarrow}$. For instance, in Figure 1(a), the set of $j = (k,l)$'s is $(1,2)$, $(1,6)$, $(1,7)$, $(2,3)$, $(2,6)$, ..., $(24,25)$. The edge notation induces the corresponding shorthand for the block-pair variables $\boldsymbol{\Sigma}_j = \boldsymbol{\Sigma}_{kl}$, $\boldsymbol{Y}_j = \boldsymbol{Y}_{kl}$, and $\boldsymbol{X}_j = \boldsymbol{X}_{kl}$ defined in (3) and (4).

For data on a regular grid, and with a regular block design, the relative distances between sites in block-pairs are the same, giving identical covariance matrices $\boldsymbol{\Sigma}_j$ for all equal-configuration block pair neighbors $j = (k,l)$, under stationarity and isotropy assumptions on the random field. In this case only a few required determinants and inverses need to be computed for every block CL evaluation. We note that other methods (in particular spectral methods) can also exploit regular grids for additional speed-ups.

2.3 Guidelines for blocking

The optimal blocking will depend on the sampled spatial locations as well as the spatial correlation model and therefore cannot be determined in general. Nonetheless, we provide guidelines on how to create the blocking structures that performed well in our experiments.

The aim is to maximize the number of blocks (for computational speed), while minimizing the correlation between observables not in a block pair (for statistical efficiency). We recommend computing the empirical variogram first, and to use this for selecting the blocks. Block widths equal to the effective spatial range is a rule of thumb, but smaller blocks also work well in our examples. The block structure could be adapted based on exploratory analysis; if preliminary analysis (e.g. directional variograms) suggests anisotropy, grids can be elongated in the direction of stronger spatial dependence. In addition, specific applications may have inherent structure that could be utilized to guide the blocking scheme. From a computational perspective, it is desirable

to have a similar number of points in each block. This is achieved by using equi-sized blocks for regularly-sampled data, while irregular designs require smaller blocks in regions with high sampling density. In our data examples we experiment with both approaches.

Given the relative insensitivity with respect to blocking choices that we have observed (Tables 1, 2, 3), iterative modification of the blocking scheme, for example based on prediction results of initial blocking structures, seems too computationally demanding to be worthwhile. We instead recommend trying different number of blocks and block sizes, and checking that the results are insensitive to this choice. In practice, we find it useful to work with hundreds to thousands of sites per block. This balances the need for fast matrix factorizations with desire for statistical efficiency. More intricate blocking designs, such as overlapping blocks or including some points outside the block (similar to suggestions in Stein et al., 2004), are nevertheless possible.

3 Inference and prediction using block composite likelihood

3.1 Properties of the Maximum Composite Likelihood Estimator

The maximum CL estimates of $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ are given by

$$(\hat{\boldsymbol{\beta}}_{CL}, \hat{\boldsymbol{\theta}}_{CL}) = \text{argmax}_{\boldsymbol{\beta},\boldsymbol{\theta}} \left[\ell_{CL}(\boldsymbol{Y}; \boldsymbol{\beta}, \boldsymbol{\theta})\right].$$

In general, the maximum CL estimators are known to be consistent and asymptotically normal under the same conditions as maximum likelihood estimators (Lindsay, 1988). In the case of spatial Gaussian processes, conditions such as those in Mardia and Marshall (1984) yield the desired asymptotic properties for the resultant maximum CL estimators (Curriero and Lele, 1999; Varin, 2008, e.g).

A useful place to begin analytical exposition is with the vector-valued block composite score function, defined as $\partial \ell_{CL}(\boldsymbol{Y}; \boldsymbol{\beta}, \boldsymbol{\theta})/\partial \theta_r$, $r = 1, \ldots, R$, where R is the dimension of the parameter $\boldsymbol{\theta}$. For notational simplicity, we assume here that $\boldsymbol{Y}$ is a mean-zero Gaussian random field. Differentiating (5) with respect to θ_r, the score (Appendix) can be expressed as

$$\frac{\partial \ell_{CL}(\boldsymbol{Y}; \boldsymbol{\theta})}{\partial \theta_r} = \sum_j \Big[-\frac{1}{2}\text{trace}\left(\boldsymbol{Q}_j \frac{\partial \boldsymbol{\Sigma}_j}{\partial \theta_r}\right) + \frac{1}{2}\boldsymbol{Y}_j^t \boldsymbol{Q}_j \frac{\partial \boldsymbol{\Sigma}_j}{\partial \theta_r} \boldsymbol{Q}_j \boldsymbol{Y}_j \Big],$$

where $\boldsymbol{Q} = \boldsymbol{\Sigma}_j^{-1}$. Taking expectations and using $\mathrm{E}(\boldsymbol{Y}^t\boldsymbol{B}\boldsymbol{Y}) = \mathrm{trace}(\boldsymbol{B}\boldsymbol{\Sigma})$ we see that

$$E\left(\frac{\partial \ell_{CL}(\boldsymbol{Y};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_r}\right) = \sum_j \Big[-\frac{1}{2}\mathrm{trace}\left(\boldsymbol{Q}_j \frac{\partial \boldsymbol{\Sigma}_j}{\partial \theta_r}\right) + \frac{1}{2}\mathrm{trace}\left(\boldsymbol{Q}_j \frac{\partial \boldsymbol{\Sigma}_j}{\partial \theta_r}\right) \Big] = 0,$$

revealing that the block composite score is an unbiased estimating function for $\boldsymbol{\theta}$ for any blocking scheme.

As is typical of asymptotically-normal estimators resulting from unbiased estimating functions, the asymptotic covariance of $\hat{\boldsymbol{\theta}}_{CL}$ under expanding domain asymptotics has a sandwich form (Godambe, 1960), and then $\hat{\boldsymbol{\theta}}_{CL} \sim N(\boldsymbol{\theta}, \boldsymbol{G}^{-1})$ under suitable regularity conditions (Varin, 2008). Here, the sandwich information is

$$\begin{aligned} \boldsymbol{G} = \boldsymbol{G}(\boldsymbol{\theta}) &= \boldsymbol{H}(\boldsymbol{\theta})\boldsymbol{J}^{-1}(\boldsymbol{\theta})\boldsymbol{H}(\boldsymbol{\theta}), \\ \boldsymbol{H}(\boldsymbol{\theta}) &= -E\left(\frac{\partial^2 \ell_{CL}(\boldsymbol{Y};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^2}\right), \quad \boldsymbol{J}(\boldsymbol{\theta}) = \mathrm{Var}\left(\frac{\partial \ell_{CL}(\boldsymbol{Y};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right). \end{aligned}$$

We note that in the case of the full likelihood function, $\boldsymbol{H}(\boldsymbol{\theta})\boldsymbol{J}^{-1}(\boldsymbol{\theta}) = \boldsymbol{I}$, so $\boldsymbol{G}(\boldsymbol{\theta})$ is just the Fisher information. For the block CL, analytical expressions are available for both $\boldsymbol{H}(\boldsymbol{\theta})$ and $\boldsymbol{J}(\boldsymbol{\theta})$. The negative expected Hessian (Appendix) has elements

$$H_{rs}(\boldsymbol{\theta}) = \sum_j \frac{1}{2}\mathrm{trace}\left(\boldsymbol{Q}_j \frac{\partial \boldsymbol{\Sigma}_j}{d\theta_s}\boldsymbol{Q}_j \frac{\partial \boldsymbol{\Sigma}_j}{d\theta_r}\right), \quad r, s = 1, \ldots, R. \tag{6}$$

The expression for $\boldsymbol{J}(\boldsymbol{\theta})$ is in the Appendix. In practice we evaluate $\boldsymbol{H}(\boldsymbol{\theta})$ and $\boldsymbol{J}(\boldsymbol{\theta})$ at $\hat{\boldsymbol{\theta}}_{CL}$.

To re-introduce covariates into the model, we simply substitute $(\boldsymbol{Y}_j - \boldsymbol{X}_j\hat{\boldsymbol{\beta}}_{CL})$ for $\boldsymbol{Y}_j$ into the above expressions. For fixed $\boldsymbol{\theta}$, the regression estimate $\hat{\boldsymbol{\beta}}_{CL}$ is analytically available by writing out the quadratic form in (5) in terms of $\boldsymbol{\beta}$. This gives

$$\hat{\boldsymbol{\beta}}_{CL} = \boldsymbol{A}^{-1}\boldsymbol{b}, \quad \boldsymbol{A} = \sum_j \boldsymbol{X}_j^t\boldsymbol{Q}_j\boldsymbol{X}_j, \quad \boldsymbol{b} = \sum_j \boldsymbol{X}_j^t\boldsymbol{Q}_j\boldsymbol{Y}_j. \tag{7}$$

The covariance matrix of the limiting normal distribution of $\hat{\boldsymbol{\beta}}_{CL}$ also has a sandwich form, which is computed from the expected Hessian $\boldsymbol{H}(\boldsymbol{\beta}) = \boldsymbol{A}$ and the variance of the score:

$$\begin{aligned} \boldsymbol{J} = \mathrm{Var}\left(\frac{\partial \ell_{CL}(\boldsymbol{Y};\boldsymbol{\theta})}{\partial \boldsymbol{\beta}}\right) &= \sum_j \sum_{j'} \mathrm{Cov}(\boldsymbol{X}_j^t\boldsymbol{Q}_j\boldsymbol{Y}_j, \boldsymbol{X}_{j'}^t\boldsymbol{Q}_{j'}\boldsymbol{Y}_{j'}) \\ &= \sum_j \sum_{j'} \boldsymbol{X}_j^t\boldsymbol{Q}_j\mathrm{Cov}(\boldsymbol{Y}_j, \boldsymbol{Y}_{j'})\boldsymbol{Q}_{j'}\boldsymbol{X}_{j'}. \end{aligned} \tag{8}$$

In practice, we only sum over terms with edges $j = (k, l)$ and $j' = (k', l')$ that have common nodes among the blocks (k, l, k', l').

3.2 Prediction using block composite likelihood

Suppose that we want to predict the value of $Y(\boldsymbol{s}_0)$ at an unobserved site $\boldsymbol{s}_0$. The best linear unbiased prediction of $Y(\boldsymbol{s}_0)$ given the data $\boldsymbol{Y}$ is

$$\hat{Y}(\boldsymbol{s}_0) = \boldsymbol{x}^t(\boldsymbol{s}_0)\boldsymbol{\beta} + \boldsymbol{\Sigma}_{0,1:n}\boldsymbol{\Sigma}^{-1}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}), \tag{9}$$

where $\boldsymbol{\Sigma}_{0,1:n}$ is the cross-covariance between $\boldsymbol{s}_0$ and all observation sites $\{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_n\}$. The computational difficulty is related to matrix factorization for large n. The prediction (9) is the well-known Kriging equation. It is the optimal prediction based on the Gaussian likelihood. In this section, we describe the optimal prediction based on the block CL. The composite prediction is fast to compute and avoids storing the entire dataset at once. Throughout this section, we assume the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ are known, but in practice we use plug-in values $\boldsymbol{\beta} = \hat{\boldsymbol{\beta}}_{CL}$ and $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}_{CL}$.

Consider the task of making predictions at $n_{k0} \geq 1$ unobserved sites, all situated within block k. The first step is to augment the data vector with an un-observed vector $\boldsymbol{Y}_{k0}$ at n_{k0} prediction locations $\boldsymbol{s}_0$ such that the augmented vector $\boldsymbol{Y}_k^a = (\boldsymbol{Y}_{k0}^t, \boldsymbol{Y}_k^t)^t$. By including $\boldsymbol{Y}_{k0}$ as unobserved data in the block CL and setting the derivative of ℓ_{CL} with respect to $\boldsymbol{Y}_{k0}$ equal to 0, we obtain the composite predictions $\hat{\boldsymbol{Y}}_{k0}$.

The contribution of the unobserved data $\boldsymbol{Y}_{k0}$ to the CL is given by block terms (k, l), $l \in N_k$, looking both forward and backward in the graph of block-edges. We organize these pairs such that block k is at the top in every block-pair. The $(n_{k0} + n_k + n_l) \times (n_{k0} + n_k + n_l)$ precision matrix for $(\boldsymbol{Y}_k^{a\mathrm{t}}, \boldsymbol{Y}_l^t)^t$ is denoted $\boldsymbol{Q}_{0kl} = \{\boldsymbol{Q}_{0kl}(i, j)\}$ with indices i, j for blocks 0 (prediction sites in block k), 1 (data sites in block k) and 2 (data sites in block l). The block CL at the unobserved locations

is thus

$$\ell_{CL}(\boldsymbol{Y}_{k0}) = \sum_{l\in N_k}\Big[\text{const} - \frac{1}{2}(\boldsymbol{Y}_{k0}-\boldsymbol{X}_{k0}\boldsymbol{\beta})^t\boldsymbol{Q}_{0kl}(0,0)(\boldsymbol{Y}_{k0}-\boldsymbol{X}_{k0}\boldsymbol{\beta}) \\ - (\boldsymbol{Y}_{k0}-\boldsymbol{X}_{k0}\boldsymbol{\beta})^t\boldsymbol{Q}_{0kl}(0,1)(\boldsymbol{Y}_{k}-\boldsymbol{X}_{k}\boldsymbol{\beta}) \\ - (\boldsymbol{Y}_{k0}-\boldsymbol{X}_{k0}\boldsymbol{\beta})^t\boldsymbol{Q}_{0kl}(0,2)(\boldsymbol{Y}_{l}-\boldsymbol{X}_{l}\boldsymbol{\beta})\Big], \tag{10}$$

now regarded as a function of $\boldsymbol{Y}_{k0}$, and where the $n_{k0}\times p$ matrix $\boldsymbol{X}_{k0}$ collects the explanatory variables at prediction sites in block k.

The first and second derivatives of $\ell_{CL}(\boldsymbol{Y}_{k0})$ are easily obtained by differentiating the quadratic form in (10). The first derivative is

$$\frac{d\ell_{CL}(\boldsymbol{Y}_{k0})}{d\boldsymbol{Y}_{k0}} = -\sum_{l\in N_k}\Big[\boldsymbol{Q}_{0kl}(0,0)(\boldsymbol{Y}_{k0}-\boldsymbol{X}_{k0}\boldsymbol{\beta}) + \boldsymbol{Q}_{0kl}(0,1)(\boldsymbol{Y}_{k}-\boldsymbol{X}_{k}\boldsymbol{\beta}) \\ + \boldsymbol{Q}_{0kl}(0,2)(\boldsymbol{Y}_{l}-\boldsymbol{X}_{l}\boldsymbol{\beta})\Big]. \tag{11}$$

Setting the derivative (11) equal to 0 gives the block composite prediction

$$\begin{aligned}\hat{Y}_{k0} &= \boldsymbol{X}_{k0}\boldsymbol{\beta} + \boldsymbol{A}_0^{-1}\boldsymbol{b}_0, \qquad (12)\\ \boldsymbol{A}_0 &= \sum_{l\in N_k}\boldsymbol{Q}_{0kl}(0,0),\ \boldsymbol{b}_0 = -\sum_{l\in N_k}\Big[\boldsymbol{Q}_{0kl}(0,1)(\boldsymbol{Y}_{k}-\boldsymbol{X}_{k}\boldsymbol{\beta}) + \boldsymbol{Q}_{0kl}(0,2)(\boldsymbol{Y}_{l}-\boldsymbol{X}_{l}\boldsymbol{\beta})\Big].\end{aligned}$$

Since the mean of $\boldsymbol{Y}_k$ is $\boldsymbol{X}_k\boldsymbol{\beta}$, for any k, it is easily seen that (11) is an unbiased estimating function for $\boldsymbol{Y}_{k0}$ by checking that $E\left(d\ell_{CL}(\boldsymbol{Y}_{k0})/d\boldsymbol{Y}_{k0}\right) = 0$. Here, the expectation is taken over $\boldsymbol{Y}$, including the random $\boldsymbol{Y}_{k0}$. Given $\hat{Y}(\boldsymbol{s}_0)$ on a fine grid of prediction locations $\boldsymbol{s}_0$, the residual spatial surface is estimated using $\hat{Y}(\boldsymbol{s}_0) - \boldsymbol{x}^t(\boldsymbol{s}_0)\boldsymbol{\beta}$, assuming we know the covariates at $\boldsymbol{s}_0$. At data locations the non-structured error residual is $Y(\boldsymbol{s}) - \hat{Y}(\boldsymbol{s})$.

The asymptotic variance of the composite prediction is described by a Godambe sandwich;

$$\begin{aligned}\boldsymbol{G}_0(\boldsymbol{Y}_{k0}) &= \boldsymbol{H}_0(\boldsymbol{Y}_{k0})\boldsymbol{J}_0^{-1}(\boldsymbol{Y}_{k0})\boldsymbol{H}_0(\boldsymbol{Y}_{k0}), \qquad (13)\\ \boldsymbol{H}_0(\boldsymbol{Y}_{k0}) &= -E\left(\frac{d^2\ell_{CL}(\boldsymbol{Y}_{k0})}{d\boldsymbol{Y}_{k0}^2}\right),\ \boldsymbol{J}_0(\boldsymbol{Y}_{k0}) = \text{Var}\left(\frac{d\ell_{CL}(\boldsymbol{Y}_{k0})}{d\boldsymbol{Y}_{k0}}\right).\end{aligned}$$

The prediction variances at locations $\boldsymbol{s}_0$ are the diagonal elements of $\boldsymbol{G}_0^{-1}(\boldsymbol{Y}_{k0})$. The Hessian is the fixed quantity $d^2\ell_{CL}(\boldsymbol{Y}_{k0})/d\boldsymbol{Y}_{k0}^2 = -\boldsymbol{A}_0$ defined in (12). The variance that defines $\boldsymbol{J}_0(\boldsymbol{Y}_{k0})$

is computed over $\boldsymbol{Y}$, including the random $\boldsymbol{Y}_{k0}$. By introducing the $n_{k0} \times (n_{k0} + n_k)$ matrix $\boldsymbol{B}_{k0} = \left[\sum_{l \in N_k} \boldsymbol{Q}_{0kl}(0,0), \sum_{l \in N_k} \boldsymbol{Q}_{0kl}(0,1)\right]$ from (11) we get

$$\begin{aligned} \boldsymbol{J}_0(\boldsymbol{Y}_{k0}) = & \quad \text{Var} \quad \left(\frac{d\ell_{CL}(\boldsymbol{Y}_{k0})}{d\boldsymbol{Y}_{0,k}} \right) = \boldsymbol{B}_{k0} \text{Var}(\boldsymbol{Y}_k^a) \boldsymbol{B}_{k0}^t \\ & + \quad 2 \sum_{l \in N_k} \boldsymbol{B}_{0k} \text{Cov}(\boldsymbol{Y}_k^a, \boldsymbol{Y}_l) \boldsymbol{Q}_{0kl}^t(0,2) \\ & + \quad \sum_{l \in N_k} \sum_{l' \in N_k} \boldsymbol{Q}_{0kl}(0,2) \text{Cov}(\boldsymbol{Y}_l, \boldsymbol{Y}_{l'}) \boldsymbol{Q}_{0kl'}^t(0,2). \end{aligned} \tag{14}$$

The derivations of the sampling properties of the composite prediction contained in this section are computed from a somewhat different standpoint than the analogous derivations for parameter estimation above. In Section 3.1, the parameters were considered fixed and unknown, while in this section we have considered the prediction variable as a random quantity. This distinction resembles that between confidence intervals and prediction intervals in traditional regression analysis.

We note that if we fix the block boundaries and allow the data density to increase to infinity, the block CL prediction converges to the Kriging prediction, and thus enjoys the same infill asymptotic properties as the optimal predictor.

4 Computation

4.1 Computing the block composite estimator

Optimization of the block CL proceeds iteratively, alternately solving for $\hat{\boldsymbol{\beta}}_{CL}$ given $\hat{\boldsymbol{\theta}}_{CL}$, and for $\hat{\boldsymbol{\theta}}_{CL}$ given $\hat{\boldsymbol{\beta}}_{CL}$. While each optimization with respect to the regression parameters $\boldsymbol{\beta}$ can be expressed analytically (see (7)), the optimization for $\boldsymbol{\theta}$ must be done numerically. We define a starting value $\boldsymbol{\theta}(0)$ and use Fisher-scoring updates according to

$$\boldsymbol{\theta}(b+1) = \boldsymbol{\theta}(b) - E\left[\frac{\partial^2 \ell_{CL}(\boldsymbol{Y}; \boldsymbol{\beta}, \boldsymbol{\theta}(b))}{\partial \boldsymbol{\theta}^2} \right]^{-1} \frac{\partial \ell_{CL}(\boldsymbol{Y}; \boldsymbol{\beta}, \boldsymbol{\theta}(b))}{\partial \boldsymbol{\theta}}. \tag{15}$$

The score $\partial \ell_{CL}(\boldsymbol{Y}; \boldsymbol{\beta}, \boldsymbol{\theta}) / \partial \boldsymbol{\theta}$ and the expectation of the second derivative of the block CL can be derived analytically (Section 3.1 and Appendix). Convergence typically occurs in about 5 Fisher-scoring updates. The expressions in (15) require the derivatives of the covariance function. For

most covariance models in common use, the derivatives $\partial \boldsymbol{\Sigma}_j / \partial \theta_r$ are available in closed form. For instance, the exponential covariance function $\Sigma_j(i, i') = \tau^2 I(i = i') + \sigma^2 \exp(-\phi h)$, $h = \|\boldsymbol{s}_i - \boldsymbol{s}_{i'}\|$, has derivatives

$$\frac{\partial \Sigma_j(i,i')}{\partial \sigma^2} = \exp(-\phi h), \ \frac{\partial \Sigma_j(i,i')}{\partial \phi} = -h\sigma^2 \exp(-\phi h), \ \frac{\partial \Sigma_j(i,i')}{\partial \tau^2} = I(i = i').$$

An efficient algorithm for the Fisher-scoring update is given in Algorithm 1. Note that many of the derivatives and matrix factorizations are the same for the score and Hessian, and hence can be re-used.

Algorithm 1 Computation for a Fisher scoring update for the block CL

Require: $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_R)^t$, initialize $u_r = 0$ and $H_{rs} = 0, r = 1, \ldots, R, s = r, \ldots, R$.

for $k = 1$ to $M - 1$ **do**
  for $l \in N_k^{\rightarrow}$ **do**
    Build and factorize $\boldsymbol{\Sigma}_{kl} = \boldsymbol{L}_{kl}\boldsymbol{L}_{kl}^t$. $\boldsymbol{Q}_{kl} = \boldsymbol{\Sigma}_{kl}^{-1} = \boldsymbol{L}_{kl}^{-t}\boldsymbol{L}_{kl}^{-1}$
    Compute $\boldsymbol{q}_{kl} = \boldsymbol{Q}_{kl}(\boldsymbol{Y}_{kl} - \boldsymbol{X}_{kl}\boldsymbol{\beta})$
    for $r = 1$ to R **do**
      Compute $\boldsymbol{W}_{klr} = \boldsymbol{Q}_{kl}\frac{d\boldsymbol{\Sigma}_{kl}}{d\theta_r}$
      $u_r \leftarrow u_r - \frac{1}{2}\text{trace}(\boldsymbol{W}_{klr}) + \frac{1}{2}q_{kl}^t \frac{d\boldsymbol{\Sigma}_{kl}}{d\theta_r} q_{kl}$
      for $s = r$ to R **do**
        $H_{rs} \leftarrow H_{rs} + \frac{1}{2}\text{trace}(\boldsymbol{W}_{klr}\boldsymbol{W}_{kls})$
      end for
    end for
  end for
end for

return $\frac{d\ell_{CL}}{d\boldsymbol{\theta}} = (u_1, \ldots, u_R)^t$, $-E\left(\frac{d^2\ell_{CL}}{d\boldsymbol{\theta}_r d\theta_s}\right) = \begin{pmatrix} H_{11} & \cdots & H_{1R} \\ \vdots & \ddots & \vdots \\ H_{1R} & \cdots & H_{RR} \end{pmatrix}$,

$\boldsymbol{\theta} = \boldsymbol{\theta} + \boldsymbol{H}^{-1}\boldsymbol{u}$.

4.2 Computational efficiency and parallel computing

To study computational aspects of the block CL approach, we compare computing times for a variety of sites per block, $n_k = c$, and data sizes $n = cM$. Under the assumption of fixed c and increasing n, the computational complexity of the block CL is $O(n)$. This follows since the for-loop goes over $n|N_k^{\rightarrow}|/c$ steps, and at every step the computation time is $O(c^3)$ for the smaller (fixed size c) matrix factorization. This kind of linear order in n is usually required for

massive datasets. It holds for most basis representations such as fixed rank Kriging (Cressie and Johannesson, 2008). In contrast, Fourier approximations are $O(n \log n)$, while Gaussian Markov random fields (Lindgren et al., 2011) are $O(n^{3/2})$ for two-dimensional spatial data and $O(n^2)$ for three spatial dimensions, unless one embeds the process in a low-rank representation.

In addition to the order-n computational complexity, the block CL approach has no limit on data size due to computer memory. Since the CL, score, and Hessian computations are sums over independent calculations for each pair of blocks, the only in-memory information is that pertinent to the current pair.

The block CL approach is highly amenable to parallelization. First, the CL expression is a sum over independent calculations for each pair of blocks, and these can be performed in parallel. Second, the main computational cost is due to linear algebra subroutines, e.g. matrix decompositions, which are also highly parallelizable (Galoppo et al., 2005; Volkov and Demmel, 2008). We investigate the use of GPUs to accelerate computation and allow for analysis of large data sets. The two approaches we assessed were a MATLAB toolbox called Jacket and CUDA (Compute Unified Device Architecture) C. Both approaches require CUDA-capable NVIDIA GPUs.

5 Simulation study

5.1 Inference and prediction

In this synthetic data example we vary the number of spatial blocks to investigate the way blocking schemes trade off statistical for computational efficiency. We generate a spatial design with $n = 2,000$ observation sites on a spatial domain $(0,1) \times (0,1)$. The selected design is of a regular plus random infill type (Diggle and Lophaven, 2006). We generate $26^2 = 676$ regular points over the domain, then select 100 of these at random and draw 10 random points around each of them from a $U(-0.04, 0.04)$ in both coordinates. The remaining 324 sites are drawn randomly within the unit square. We define 500 prediction sites, drawn from $U(0,1)$ in both coordinates, and not included in the $n = 2,000$ data.

With n as small as $2,000$ we can compare CL with full likelihood results. Covariates are

$\boldsymbol{x}^t(\boldsymbol{s}_i) = (1, s_{i1})$, with true regression parameters $\boldsymbol{\beta} = (-1, 1)^t$. We use a Matérn covariance model with smoothness parameter 3/2, i.e. $\Sigma(i, i') = \tau^2 I(h = 0) + \sigma^2(1 + \phi h)\exp(-\phi h)$, $h = ||\boldsymbol{s}_i - \boldsymbol{s}_{i'}||$.

We use a parametrization on the real line, with log precisions and log range parameter: $\theta_1 = -\log(\sigma^2)$, $\theta_2 = \log(\phi)$, and $\theta_3 = -\log(\tau^2)$. This parametrization makes the Fisher-scoring optimization robust. The scale parameters used to generate the data are $\theta_1 = 0$, $\theta_2 = 3$ and $\theta_3 = 2$. The so-called effective range, the distance at which the correlation decays to 5%, is 0.24. The asymptotic normality for parameter estimation essentially relies on expanding domain asymptotics (Cressie, 1993), which holds well when the correlation range of the process is small compared to the spatial domain (Zhang and Zimmerman, 2005). Here, the range is 1/4 of the domain, which is reasonably large in most contexts.

The results of mean square error (MSE), asymptotic relative efficiency (RE), coverage probabilities and computing time are given in Table 1. For the CL, we use regular blocks (Figure 1(a)) of lattice size $9 = 3^2$, $25 = 5^2$, $49 = 7^2$ and $100 = 10^2$. The results are averages over $1,000$ replicates of $n = 2,000$ data for the same spatial design. The asymptotic RE is defined by the ratio of the asymptotic variances obtained by the Hessian (likelihood) and the Godambe sandwich (CL).

For all models the increase in MSE for the block CL models is small relative to the full likelihood model, in particular for predictions. The asymptotic RE (in parentheses) show that standard deviations of the parameter estimates are larger for the CL models. For all block CL models, the prediction efficiency is near 1. The coverage probabilities are close to the nominal level, and not much smaller for the block CL models than for full likelihood. The slight under-coverage seen in both the CL and the full likelihood is caused by the imperfect accuracy of the asymptotic approximation to the finite sample problem, rather than by the omission of terms in the CL expression. Notably, the prediction coverages are excellent in all cases, including all composite models. The computing time is reduced by a factor 7 when using the 49 or 100 block CL models instead of the full likelihood.

We next study the performance by cross-plotting the CL and likelihood results. In Figure 2(a) we show the parameter estimates for log precision (left), log range (middle) and log nugget

Table 1: Synthetic data with Matérn (3/2) covariance function ($n = 2,000$). The asymptotic CL variances, relative to full likelihood (L), are shown in parentheses. Results are averages over $1,000$ replicates.

	L	CL9	CL25	CL49	CL100
MSE $\hat{\beta}_1$ (Asymp. RE)	0.05 (1)	0.05 (0.81)	0.06 (0.77)	0.06 (0.74)	0.06 (0.79)
MSE $\hat{\beta}_2$ (Asymp. RE)	0.17 (1)	0.21 (0.82)	0.24 (0.77)	0.23 (0.74)	0.23 (0.79)
MSE $\hat{\theta}_1$ (Asymp. RE)	0.014 (1)	0.018 (0.80)	0.019 (0.78)	0.019 (0.80)	0.018 (0.85)
MSE $\hat{\theta}_2$ (Asymp. RE)	0.0036 (1)	0.0043 (0.80)	0.0048 (0.77)	0.0052 (0.76)	0.0054 (0.76)
MSE $\hat{\theta}_3$ (Asymp. RE)	0.0007 (1)	0.0008 (0.86)	0.0008 (0.88)	0.0008 (0.88)	0.0008 (0.87)
Coverage (0.95) $\hat{\beta}_1$	0.96	0.95	0.96	0.96	0.96
Coverage (0.95) $\hat{\beta}_2$	0.93	0.92	0.93	0.92	0.92
Coverage (0.95) $\hat{\theta}_1$	0.93	0.92	0.92	0.91	0.91
Coverage (0.95) $\hat{\theta}_2$	0.94	0.94	0.93	0.91	0.91
Coverage (0.95) $\hat{\theta}_3$	0.95	0.95	0.95	0.95	0.94
MSPE (Mean Asymp. RE)	193 (1)	195 (1)	198 (1)	200 (0.99)	204 (0.97)
Mean coverage (0.95)	0.95	0.95	0.95	0.95	0.95
Computing time (sec), no GPU	76	39	16	12	12

precision (right) using maximum likelihood (x-axis) versus that of CL with 100 blocks (y-axis). The points fall near the straight line with unit slope, showing that the maximum CL estimates are very close to the maximum likelihood estimates. In Figure 2(b) we similarly show the asymptotic standard deviations based on the Hessian of the likelihood and the Godambe sandwich for the 100-block CL model. The points are above the straight line, visualizing the slight decrease in efficiency when using the block CL.

Figure 2(c) shows cross-plots of predictions, which fall tightly along the straight line with unit slope. Figure 2(d) shows the associated prediction standard errors. The points are near the straight line, indicating that the block CL model does not lose much prediction efficiency. The clusters of points off above the $y = x$ line in Figure 2(d) correspond to predictions near the block boundaries. The increased prediction variance in these regions is caused by edge effects, where the CL ignores some of the dependency effects outside the block-pair interactions. The sandwich standard errors correctly account for this effect.

The results show that the statistical efficiency decreases moderately with increased number of

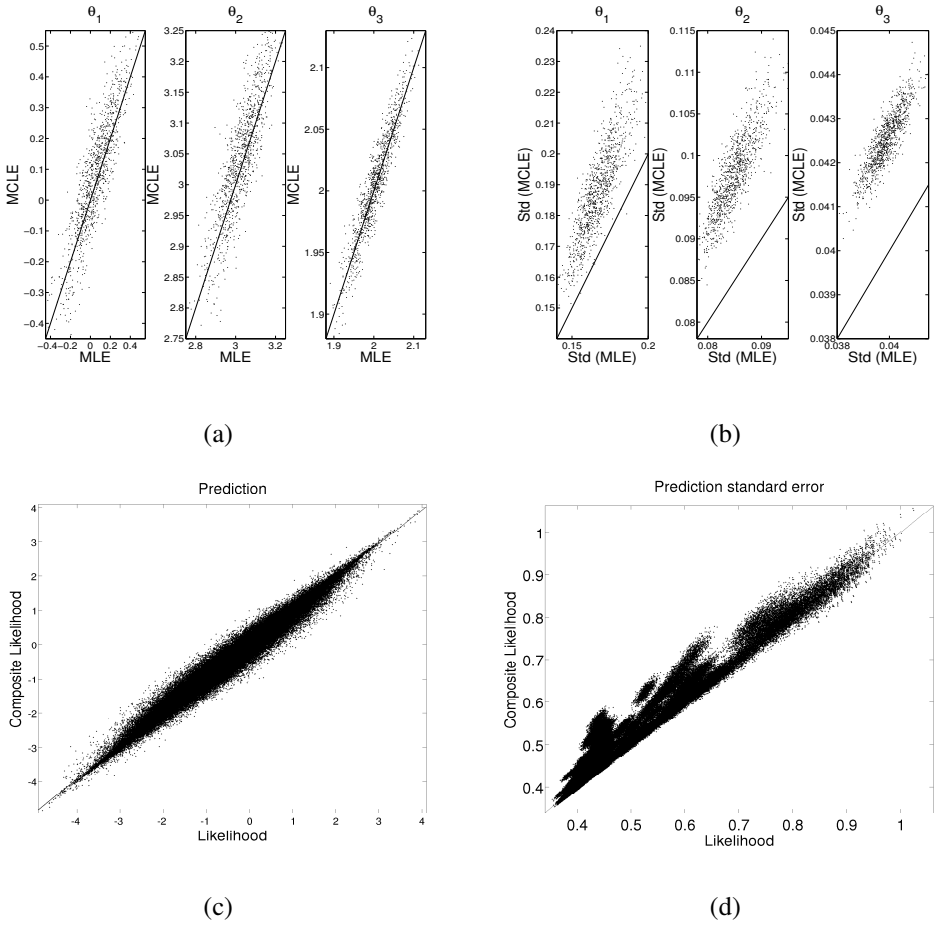


Figure 2: Synthetic data: Comparison of full likelihood (x-axis) and CL (y-axis). (a) Estimates. (b) Asymptotic standard deviation of estimators. (c) Predictions. (d) Prediction standard deviations. The CL is based on 100 blocks. The number of prediction sites is 500, and results are computed over $1,000$ replicates of data at $n = 2,000$ observation sites. θ_1 is a precision parameter, θ_2 is a range parameter and θ_3 is a nugget precision parameter.

blocks. For prediction purposes the effect is very small. We also tried other covariance functions, and the type of correlation model does not seem to affect the results much. Increasing the effective correlation length while keeping the domain fixed degrades the accuracy of the inference for parameter estimation because it decreases fidelity to the expanding domain asymptotics. This effect is largest for the spatial range parameter, while predictive performance remains quite good.

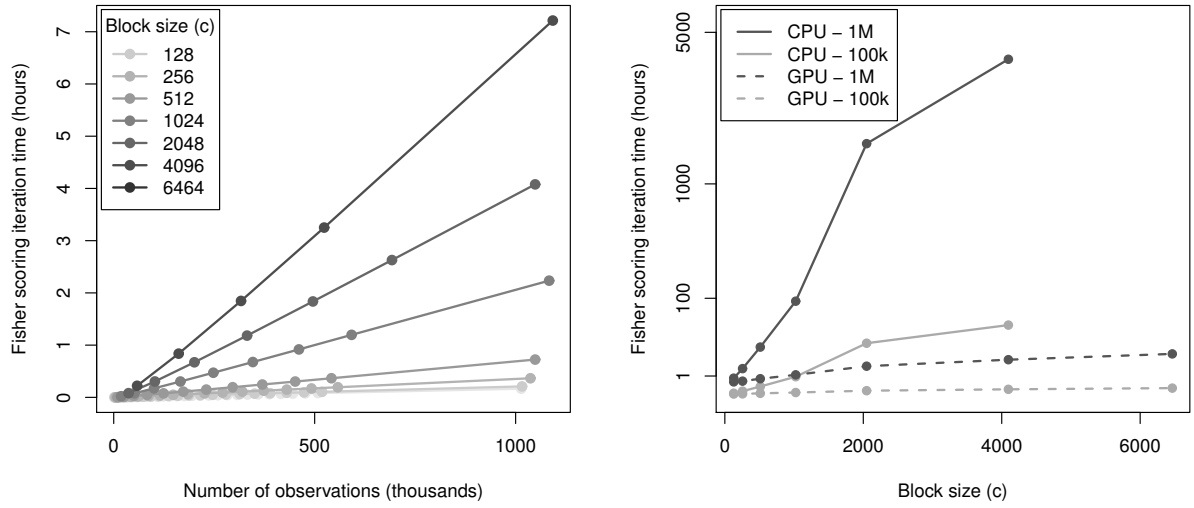
5.2 Computing time on the GPU

To assess the computational aspects of the CL algorithm in parallel computing environments, we perform simulation studies with varying block sizes and numbers of observations and record the total run time. For easier comparison we fix the number of points in the blocks $n_k = c$, for all k. This entails a spatial design with c random sites in every block. We study performance for various c and data sizes n. We again use regular blocks covering the domain $(0, 1) \times (0, 1)$, and a Matérn (3/2) covariance function with effective correlation range of about 0.24.

First, we instruct MATLAB/Jacket to utilize the `gfor` loop on the GPU. The best performance gain we attain using `gfor` is less than ten-fold relative to MATLAB code without Jacket. It may be possible to increase these gains with a thorough understanding of how Jacket interfaces with the GPU, and the associated memory allocation.

Instead, we study the computational gain when using parallel computing for matrix decomposition using CUDA C. We next show the resulting CUDA C computing times for the Fisher-scoring algorithm on synthetic datasets of varying sizes and CL models. The number of points per blocks c ranges from 128 to 4096 on the CPU, the largest block size available on a 32-bit machine, and 6464 on the GPU, the largest available block size within the GPU memory constraints, and for different dimension n. Thus, using quadratic regular grids of blocks, the smallest dataset has $128 \times 3 \times 3 = 1152$ observations and the largest dataset has $6464 \times 13 \times 13 = 1,092,416$ observations.

Figure 3(a) shows GPU computation times as a function of data size for different block sizes. This display clearly shows the linear scaling of the algorithm with data size for fixed number of points c per block. Figure 3(b) shows the associated computation times for fixed data sizes and varying block sizes on both the CPU and GPU. The computation times are plotted on a cube-root scale to emphasize that the Fisher scoring algorithm has cubic complexity in c, the number of points per block. The speed-up when running the equivalent algorithm on the GPU compared with the CPU is linear in block size and essentially constant within a given block size. The speedup was 1.4-fold at $c = 128$, 13-fold at $c = 512$, 29-fold at $c = 1024$, and 112-fold at $c = 4096$. In our experience, the Fisher scoring algorithm takes about 5 iterations to reach convergence. This



(a) (b)

Figure 3: (a) Computation times for a single Fisher scoring iteration on a NVIDIA C2050 GPU as a function of data size for a variety of block sizes. (b) Computation times, plotted on a cube-root scale, on both the CPU and GPU for a single Fisher scoring iteration as a function of block size for specific data sizes.

speed-up would allow a one million observation dataset to be fully analyzed in half a day using $c = 1024$ or about two days using $c = 4096$. Of course, one could also use parameter estimates from smaller block sizes as starting values for larger blocks, etc.

A more sophisticated implementation on the GPU would allow more speed-up for the CL model, utilizing a parallel for-loop and running matrix decompositions in parallel. This is future work.

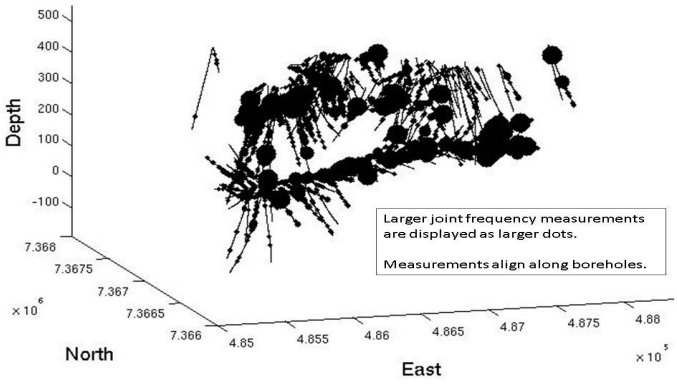
6 Real data examples

To study the performance of the block CL in real-world settings, we test it on one dataset from the mining industry and one of total column ozone generated from satellite retrievals. For the mining dataset we compare CL with the predictive process model which easily accommodates

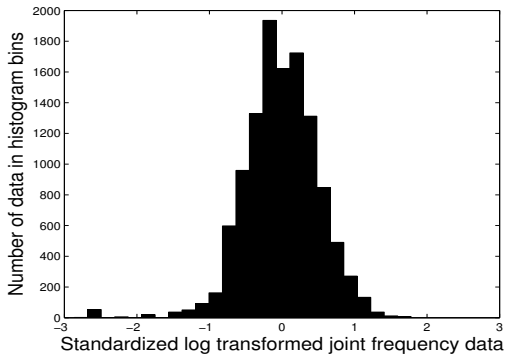
three-dimensional data. For the massive satellite dataset, we compare block CL with fixed rank Kriging, which has been used previously to analyze this dataset (Cressie and Johannesson, 2008).

6.1 Mining dataset

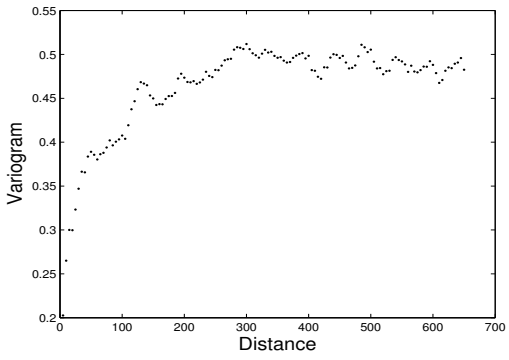
We study a joint frequency dataset acquired in an iron mine in Norway. Such borehole data are useful for predicting the stability requirements in the mine and for avoiding rock-fall. The raw data are aggregated to 4m blocks along the boreholes, and the total number of measurements is $n = 11,107$. Ellefmo and Eidsvik (2009) analyzed a subset of this dataset.



(a)



(b)



(c)

Figure 4: Mining data: (a) Data sites in a Norwegian iron mine. The data size is $n = 11,107$, collected in about 200 boreholes. (b) Histogram of shifted log transformed data of joint frequencies. (c) Empirical variogram.

In Figure 4(a) we display the three dimensional locations of the measurements. The logarithm of the joint frequency observations are standardized (Figure 4(b)). We apply a geostatistical model to the resulting data. Figure 4(c) displays the empirical variogram, with nugget about τ^2 = 0.2/2 = 0.32^2, the correlation range about 100 m, and the variance of the structured effect about σ^2 = (0.5-0.2)/2 = 0.39^2.

We use the block CL model with different block sizes. The blocks are constructed by a Voronoi / Delaunay tessellation for the (north,east) coordinates of the data, with cells extending for all depths. The tessellation is made by random sampling, without replacement, among all data sites, which on average gives smaller area blocks where sampling locations are dense. We compare the CL results with the predictive process model (with 500 or 1000 knots) using tapering of the residual covariance process (Sang and Huang, 2012). The predictive process is a linear (Kriging) combination of the observed data at the knots draw at random, without replacement, from among the data locations. We use a Wendland taper with tapering length set to 100 m.

Table 2 shows the parameter estimates, the average mean squared prediction error (MSPE) and coverage probabilities for a hold-out set of 1000 prediction sites. We compare two common covariance functions: the Cauchy(3) which is $\Sigma(h) = \sigma^2(1+\phi h)^{-3}+\tau^2 I(h=0)$, and Matérn(3/2) with $\Sigma(h) = \sigma^2(1+\phi h)\exp(-\phi h)+\tau^2 I(h=0)$. The parameter estimates are very similar for all models, but the range parameter ϕ is a little smaller for the predictive process models, imposing a smoother process. The MSPE is smaller for the CL models than for the predictive process models. The coverage probabilities are excellent for all models considered. There are only small differences between the two spatial covariance functions. The CL approach with 200 blocks is about 5 times faster than using 10 blocks, with only a slight increase in MSPE.

6.2 Total column ozone dataset

We next analyze total column ozone (TCO) data acquired from an orbiting satellite mounted with a passive sensor registering backscattered light. The dataset we consider here was previously analyzed by Cressie and Johannesson (2008), and is displayed in Figure 5. The dataset consists of $n = 173,405$ measurements. Cressie and Johannesson (2008) used fixed rank Kriging (FRK)

Table 2: Mining data: Parameter estimates, MSPE and coverage probabilities for prediction distributions. The different columns correspond to different number of blocks for the CL model and different knot sizes for the predictive process models with tapered residuals (PP+T).

		CL, 200	CL, 40	CL, 10	PP+T, 500	PP+T, 1000
Cauchy	$\hat{\sigma}$	0.41	0.42	0.42	0.41	0.41
	$\hat{\phi}$	0.013	0.013	0.012	0.010	0.011
	$\hat{\tau}$	0.29	0.29	0.29	0.29	0.29
	MSPE	145	144	143	216	189
	Pred cov (0.95)	0.95	0.95	0.95	0.95	0.95
	Timing (min)	1	2	5	8	30
Matérn (3/2)	$\hat{\sigma}$	0.38	0.39	0.39	0.39	0.39
	$\hat{\phi}$	0.071	0.070	0.068	0.061	0.063
	$\hat{\tau}$	0.33	0.33	0.33	0.33	0.33
	MSPE	149	148	148	217	188
	Pred cov (0.95)	0.95	0.95	0.95	0.95	0.95
	Timing (min)	1	2	5	8	30

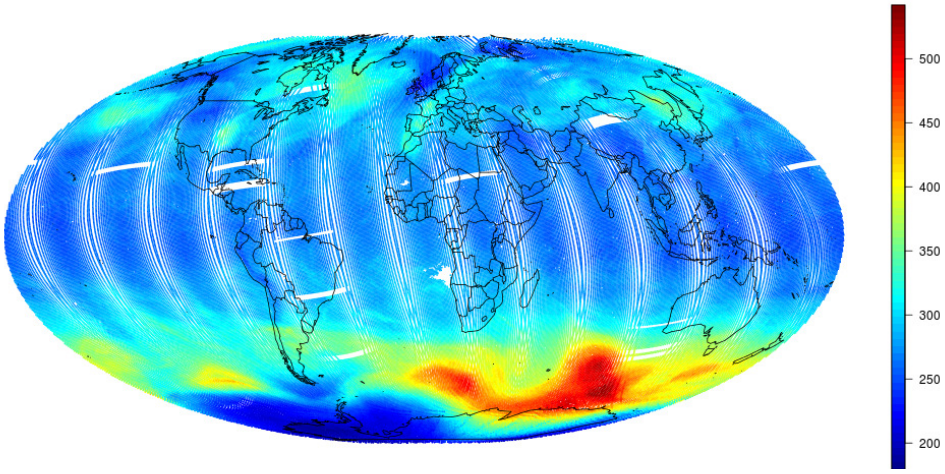


Figure 5: Total column ozone data: Number of measurements is $n = 173,405$.

in their analysis. This approach is based on a basis representation of the spatial Gaussian process. They use 396 local bi-square basis functions at various resolutions recovered from a discrete global grid (Sahr, 2011).

We compare the block CL models using 15, 24 and 30 regular latitude / longitude blocks. For

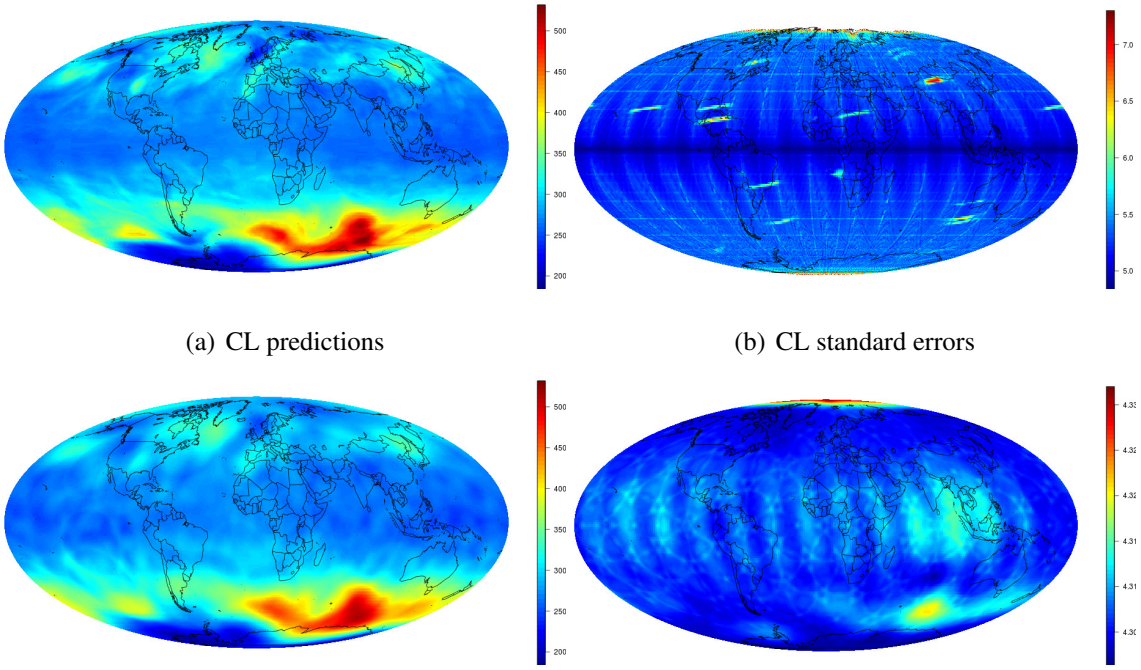
the one with most blocks (900), we only use the four nearest neighbors in the CL expression. The 15 and 24 cases have eight neighbors. We use a fixed and constant mean β_1, and a Cauchy (3) type covariance function. The block CL parameter estimates are similar for the three models: $(\hat{\sigma}^2, \hat{\phi}, \hat{\tau}^2)$ is $(70^2, 0.011, 4.67^2)$ using 15×15 blocks, $(67^2, 0.009, 4.67^2)$ using 24×24 blocks, and $(63^2, 0.007, 4.69^2)$ for the nearest neighbor blocking scheme.

We predict on a 180×288 grid, corresponding to the so-called NASA level 2 data product. The latitude ranges from -89.5 to 89.5 in 1° steps, while longitude ranges from -179.375 to 179.375 in 1.25° steps. In total, this entails $51,840$ prediction sites. Prediction maps of TCO for the three different block CL models are very similar. Figure 6(a) shows the prediction map of TCO using 30×30 blocks. The marginal prediction standard deviations of TCO are displayed in Figure 6(b). We notice that the prediction standard deviations are much higher near the Arctic because there is no data there. We further note the increased estimated uncertainty in regions of missing data, and light-colored lines going south-southwest, where there is less dense satellite coverage. In addition, there are visible artifacts of the block CL model in the prediction standard deviations. These regions of increased estimated uncertainty where data is lacking and on the border of spatial blocks are desirable—indeed, they indicate that the sandwich variance calculations are correctly accounting for sparse data and block boundary effects. As is also desirable, these block boundary effects are not seen on the prediction map.

We next compare the CL with FRK. Here, we follow Cressie and Johannesson (2008) in using multi-resolution bi-square basis functions centered at the 3 lowest resolution discrete global grids. Resolution 4 is used to construct the binning for the moment-based parameter estimation approach (Cressie and Johannesson, 2008). Figure 6 shows the FRK predictions (c) and prediction standard deviations (d) with the nugget effect properly accounted for. Notably, the predictions obtained by FRK are much smoother than the block CL results. Moreover, the estimated prediction standard deviations are smaller for FRK and vary less around the globe. The patches of missing data are not visible in Figure 6(d). Similar to the block edge effects in the CL model, the locations of the basis functions are easily seen as artifacts in the estimated FRK standard error map.

To compare prediction and coverage accuracy, we use a hold-out set of $25,000$ randomly-

selected data locations around the globe. We estimate the model parameters based on the remaining data and predict at the hold-out locations. Table 3 shows the comparison of the block CL and FRK. All block CL models obtain coverages close to the nominal rate and have similar prediction error. FRK with resolution 3 or 4 (using more basis functions) shows much larger prediction errors and it under-covers conspicuously. These results show that for this dataset, the low-dimensional representation in FRK is over-smoothing. The approximate computing times tell us that FRK with resolution 3 is the fastest, but CL with many blocks is comparable.



(a) CL predictions (b) CL standard errors

(c) FRK predictions (d) FRK standard errors

Figure 6: Total Column Ozone data: Top displays are based on the CL model with 30×30 longitude and latitude blocks. Bottom displays are based on fixed rank Kriging. Predictions (a & c) and prediction standard deviations (b & d).

Bolin and Lindgren (2011) also analyze the TOMS data and produce predictions that visually resemble ours. Their statistical model is quite different in that it constructs a dependence model through a stochastic differential equation formulation, whereas the CL approach works directly with the covariance function. The stochastic differential equation approach directly generates a

Table 3: Total Column Ozone data: Mean square prediction error (MSPE) and coverage probabilities (95 %) for a $25,000$ hold-out set. The results are for fixed rank Kriging with resolution 4 (FRK, 4) and resolution 3 (FRK, 3), and for CL model with 15×15 (CL, reg15), 24×24 blocks (CL, reg 24), and a four neighbor version with 30×30 blocks (CL, fast 30).

	FRK, 4	FRK, 3	CL, reg 15	CL, reg 24	CL, fast 30
MSPE	44.3	88.1	25.7	26.1	26.0
Pred cov (0.95)	0.81	0.71	0.96	0.96	0.96
Timing (min)	6	2	40	20	4

very sparse precision matrix. However, factors of this sparse matrix, required for estimation and prediction, may not be sparse, particularly for three-dimensional applications. As a result, storing this large matrix factor becomes prohibitive for large enough datasets. This restriction may be mitigated to some degree by leveraging ideas from numerical linear algebra (Aune et al., 2012).

7 Closing remarks

In this paper we use a block CL model for parameter estimation and prediction in large Gaussian spatial models. The properties of the CL are well-understood in the context of parameter estimation. Here we also present a method for spatial prediction using the block CL. We show through a simulation study that the block CL performs well for reasonably-sized blocks, especially for spatial prediction. Using the divide and conquer strategy inherent in the CL, the required computation time is reduced considerably relative to likelihood-based calculations. We also test the block CL on one large dataset from the mining industry ($n = 11,107$) and one massive dataset from satellite measurements ($n = 173,405$). For these datasets we compare the block CL with predictive process models and with fixed rank Kriging. The full-rank block CL method provides better results in terms of mean square errors and coverage probabilities in the examples we considered.

The block CL approach requires the selection of blocks. We provide guidelines for this step in the paper, and try several alternatives in the examples. In practice we recommend testing results with a couple of choices of block sizes (hundreds to thousands sites per block) and blocking

designs. The optimal blocking strategy would depend on the spatial correlation and the design of data points, and prediction results do not seem very sensitive to the detailed blocking approach.

We implemented parallel versions of the block CL model. For moderate to large block sizes this parallel implementation gives speed-ups for 2–3 orders of magnitude, on top of the speed-up achieved by the CL construction. A topic for future work is to tailor the distribution of block data to the GPU for maximum reduction of the computing time, in a software package. CPU and GPU examples of code as well as datasets are available as supplemental material at the journal's website.

In this paper we considered only spatial Gaussian processes. In future work we aim to look at spatio-temporal processes as well. For example, the satellite data in Section 6.2 is from just one day of retrievals. It is more useful to analyze these types of data over several days. Bai et al. (2012) studied composite models for parameter estimation in spatio-temporal models and Bevilacqua et al. (2012) propose weighted composite likelihood models for pairwise interactions of space-time variables.

It would also be interesting to study the current approach in hierarchical models. It is relatively straightforward to predict a latent variable, say $w(\boldsymbol{s})$ in a block k, by adding this in the data vector $(\boldsymbol{Y}_k^t, \boldsymbol{w}_k^t)$ for block k, and then computing the score, Hessian and prediction sandwich for missing data $\boldsymbol{w}_k$. Spatial generalized linear models have been studied in the CL literature (Varin et al., 2011), and we believe approximate block CL approaches may be suitable here. More complex hierarchical modeling structures are challenging in this setting.

Data dimensions will likely become even larger in the future. We foresee a larger future interest in $O(n)$ approximations for spatial and spatio-temporal applications, as well as in the use of parallel computing environments.

Supplementary material

Appendix: Score function and Hessian. (pdf file)

Datasets and CPU and GPU examples of code with 'readme'-files. (zip file)

Acknowledgements

We thank the Statistical and Applied Mathematical Sciences Institute (SAMSI) for support during the program on space-time analysis (2009-2010). We thank NVIDIA for supporting us with graphics cards. Rana Gruber provided the joints data, while Noel Cressie and Gardar Johannesson made the TCO data acquired by NASA available to us.

References

Aune, E., D. P. Simpson, and J. Eidsvik (2012). Parameter estimation in high dimensional Gaussian distributions. *Statistics and Computing. To appear*.

Bai, Y., P. X. K. Song, and T. E. Raghunathan (2012). Joint composite estimating functions in spatiotemporal models. *J. R. Stat. Soc. Ser. B Stat. Methodol. 74*(5), 799–824.

Banerjee, S., A. E. Gelfand, A. O. Finley, and H. Sang (2008). Gaussian predictive process models for large spatial data sets. *J. R. Stat. Soc. Ser. B Stat. Methodol. 70*(4), 825–848.

Bevilacqua, M., C. Gaetan, J. Mateu, and E. Porcu (2012). Estimating space and space-time covariance functions for large data sets: a weighted composite likelihood approach. *J. Amer. Stat. Assoc. 107*(497), 268–280.

Bolin, D. and F. Lindgren (2011). Spatial models generated by nested stochastic partial differential equations, with an application to global ozone mapping. *Ann. of Appl. Stat. 5*(1), 523–550.

Caragea, P. C. and R. L. Smith (2007). Asymptotic properties of computationally efficient alternative estimators for a class of multivariate normal models. *J. Multivariate Anal. 98*(7), 1417–1440.

Cressie, N. (1993). *Statistics for spatial data*. Wiley. New York: John Wiley & Sons Inc.

Cressie, N. and G. Johannesson (2008). Fixed rank kriging for very large spatial data sets. *J. Roy. Statist. Soc. Ser. B 70*(1), 209–226.

Curriero, F. C. and S. Lele (1999). A composite likelihood approach to semivariogram estimation. *J. Agric. Biol. Environ. Stat. 4*(1), 9–28.

Diggle, P. and S. Lophaven (2006). Bayesian geostatistical design. *Scand. J. Statist. 33*(1), 53–64.

Ellefmo, S. and J. Eidsvik (2009). Local and spatial joint frequency uncertainty and its application to rock mass characterisation. *Rock mechanics and rock engineering 42*(4), 667–688.

Furrer, R., M. G. Genton, and D. Nychka (2006). Covariance tapering for interpolation of large spatial datasets. *J. Comput. Graph. Statist. 15*(3), 502–523.

Galoppo, N., N. Govindaraju, M. Henson, and D. Manocha (2005). LU-GPU: Efficient algorithms for solving dense linear systems on graphics hardware. In *Proceedings of the 2005 ACM/IEEE conference on Supercomputing*, pp. 3. IEEE Computer Society.

Godambe, V. (1960). An optimum property of regular maximum likelihood estimation. *The Annals of Mathematical Statistics 31*(4), 1208–1211.

Godambe, V. P. and C. C. Heyde (1987). Quasi-likelihood and optimal estimation. *Internat. Statist. Rev. 55*(3), 231–244.

Heagerty, P. J. and S. R. Lele (1998). A composite likelihood approach to binary spatial data. *J. Amer. Statist. Assoc. 93*(443), 1099–1111.

Kaufman, C., M. Schervish, and D. Nychka (2008). Covariance tapering for likelihood-based estimation in large spatial datasets. *J. Amer. Statist. Assoc. 103*(484), 1545–1569.

Lee, A., C. Yau, M. B. Giles, A. Doucet, and C. C. Holmes (2010). On the utility of graphics cards to perform massively parallel simulation of advanced Monte Carlo methods. *J. Comput. Graph. Statist. 19*(4), 769–789.

Lindgren, F., J. Lindstrøm, and H. Rue (2011). An explicit link between gaussian fields and gaussian markov random fields: The spde approach. *J. Roy. Statist. Soc. Ser. B 73*(3), 423–498.

Lindsay, B. G. (1988). Composite likelihood methods. *Contemp. Math., Amer. Math. Soc. 80*, 221–239.

Mardia, K. V. and R. J. Marshall (1984). Maximum likelihood estimation of models for residual covariance in spatial regression. *Biometrika 71*(1), 135–146.

Sahr, K. (2011). Dggrid version 3.1b. user documentation for discrete global grid generation software. *Southern Oregon*. Available from www.sou.edu/cs/sahr/dgg.

Sang, H. and J. Z. Huang (2012). A full scale approximation of covariance functions for large spatial data sets. *J. R. Stat. Soc. Ser. B Stat. Methodol. 74*(1), 111–132.

Shaby, B. and D. Ruppert (2012). Tapered covariance: Bayesian estimation and asymptotics. *J. Comp. Graph. Statist. 21*(2), 433–452.

Stein, M. L. (2008). A modeling approach for large spatial datasets. *J. Korean Statist. Soc. 37*(1), 3–10.

Stein, M. L., Z. Chi, and L. J. Welty (2004). Approximating likelihoods for large spatial data sets. *J. R. Stat. Soc. Ser. B Stat. Methodol. 66*(2), 275–296.

Suchard, M. and A. Rambaut (2009). Many-core algorithms for statistical phylogenetics. *Bioinformatics 25*(11), 1370.

Suchard, M., Q. Wang, C. Chan, J. Frelinger, A. Cron, and M. West (2010). Understanding GPU programming for statistical computation: Studies in massively parallel massive mixtures. *J. Comput. Graph. Statist. 19*(2), 419–438.

Varin, C. (2008). On composite marginal likelihoods. *AStA Adv. Stat. Anal. 92*(1), 1–28.

Varin, C., N. Reid, and D. Firth (2011). An overview of composite likelihood methods. *Stat. Sin. 21*(1), 5–42.

Vecchia, A. V. (1988). Estimation and model identification for continuous spatial processes. *J. Roy. Statist. Soc. Ser. B 50*(2), 297–312.

Volkov, V. and J. Demmel (2008). Benchmarking GPUs to tune dense linear algebra. In *Proceedings of the 2008 ACM/IEEE conference on Supercomputing*, pp. 1–11. IEEE Press.

Zhang, H. and D. L. Zimmerman (2005). Towards reconciling two asymptotic frameworks in spatial statistics. *Biometrika 92*(4), 921–936.