Methods for Analyzing Large Spatial Data:

A Review and Comparison

Matthew J. Heaton, Abhirup Datta, Andrew Finley, Reinhard Furrer,

Raj Guhaniyogi, Florian Gerber, Robert B. Gramacy, Dorit Hammerling,

Matthias Katzfuss, Finn Lindgren, Douglas W. Nychka,

Furong Sun and Andrew Zammit-Mangion

October 16, 2017

AUTHOR FOOTNOTE: Matthew J. Heaton is Assistant Professor, Department of Statistics, Brigham Young University 223 TMCB, Provo, UT 84602 (email: mheaton@stat.byu.edu); Abhirup Datta is Assistant Professor, Department of Biostatistics, Johns Hopkins University, 615 N. Wolfe Street, Baltimore, MD 21205 (email: abhidatta@jhu.edu); Andrew Finley is Associate Professor, Department of Forestry and Geography, Michigan State University, 126 Natural Resources Building, Each Lansing, MI 48824 (email: finleya@msu.edu); Reinhard Furrer is Professor, Department of Mathematics and Department of Computational Science, University of Zurich, Switzerland (email: reinhard.furrer@math.uzh.ch); Rajarshi Guhaniyogi is Assistant Professor, Department of Applied Mathematics & Statistics, University of California Santa Cruz, 1156 High Street, SOE2, Santa Cruz, CA 95064 (email: rguhaniy@ucsc.edu); Florian Gerber is postdoctoral research fellow, Department of Mathematics, University of Zurich, Switzerland (email: florian.gerber@math.uzh.ch); Robert B. Gramacy is Professor, Department of Statistics, Virginia Tech, Department of Statistics (MC0439), Hutcheson Hall, 250 Drillfield Drive, Blacksburg, VA 24061 (email: rbg@vt.edu); Dorit Hammerling is Section Leader of Statististics and Data Science, Institute for Mathematics Applied to Geosciences, National Center for Atmospheric Research, Boulder, CO 80305 (email: dorith@ucar.edu); Matthias Katzfuss is Assistant Professor, Department of Statistics, Texas A&M University, College Station, TX 77843 (email: katzfuss@stat.tamu.edu); Finn Lindgren is Chair of Statis-

tics, School of Mathematics, University of Edinburgh, Edinburgh EH9 3FD, United Kingdom (email: finn.lindgren@ed.ac.uk); Douglas Nychka is Director, Institute for Mathematics Applied to Geosciences, National Center for Atmospheric Research, Boulder, CO 80305 (email: nychka@ucar.edu); Furong Sun is graduate student, Department of Statistics, Virginia Tech, Hutcheson Hall, 250 Drillfield Drive, Blacksburg, VA 24061 (email: furongs@vt.edu). Andrew Zammit-Mangion is Senior Lecturer, School of Mathematics and Applied Statistics, University of Wollongong, NSW 2522, Australia (email: azm@uow.edu.au). This material is based upon work supported by the National Science Foundation (NSF) under Grant Number DMS-1417856. Dr. Katzfuss is partially supported by NSF Grants DMS–1521676 and DMS–1654083. Dr. Gramacy and Furong Sun are partially supported by NSF Award #1621746. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

ABSTRACT: The Gaussian process is an indispensable tool for spatial data analysts. The onset of the "big data" era, however, has lead to the traditional Gaussian process being computationally infeasible for modern spatial data. As such, various alternatives to the full Gaussian process that are more amenable to handling big spatial data have been proposed. These modern methods often exploit low rank structures and/or multi-core and multi-threaded computing environments to facilitate computation. This study provides, first, an introductory overview of several methods for analyzing large spatial data. Second, this study describes the results of a predictive competition among the described methods as implemented by different groups with strong expertise in the methodology. Specifically, each research group was provided with two training datasets (one simulated and one observed) along with a set of prediction locations. Each group then wrote their own implementation of their method to produce predictions at the given location and each which was subsequently run on a common computing environment. The methods were then compared in terms of various predictive diagnostics. Supplementary materials regarding implementation details of the methods and code are available for this article online.

KEY WORDS: Big data; Gaussian process; Parallel computing; Low rank approximation.

---

1. INTRODUCTION

For decades, the Gaussian process (GP) has been the primary tool used for the analysis of geostatistical (point-referenced) spatial data (Schabenberger and Gotway 2004, Cressie 1993, Cressie

and Wikle 2015, Banerjee et al. 2014). A spatial process $Y(\boldsymbol{s})$ for $\boldsymbol{s} \in \mathcal{D} \subset \mathbb{R}^2$ is said to follow a GP if any realization $\boldsymbol{Y} = (Y(\boldsymbol{s}_1), \ldots, Y(\boldsymbol{s}_N))'$ at the finite number of locations $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N$ follows an N-variate Gaussian distribution. More specifically, let $\mu(\boldsymbol{s}) : \mathcal{D} \to \mathbb{R}$ denote a mean function returning the mean at location $\boldsymbol{s}$ (typically assumed to be linear in covariates $\boldsymbol{X}(\boldsymbol{s}) = (1, X_1(\boldsymbol{s}), \ldots, X_P(\boldsymbol{s}))'$) and $\mathbb{C}(\boldsymbol{s}_1, \boldsymbol{s}_2) : \mathcal{D}^2 \to \mathbb{R}^+$ denote a positive definite covariance function. Then, if $Y(\boldsymbol{s})$ follows a spatial Gaussian process, $\boldsymbol{Y}$ has the density function,

$$f_{\boldsymbol{Y}}(\boldsymbol{y}) = \left(\frac{1}{\sqrt{2\pi}}\right)^N |\boldsymbol{\Sigma}|^{-1/2} \exp\left\{-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{y}-\boldsymbol{\mu})\right\} \tag{1}$$

where $\boldsymbol{\mu} = (\mu(\boldsymbol{s}_1), \ldots, \mu(\boldsymbol{s}_N))'$ is the mean vector and $\boldsymbol{\Sigma} = \{\mathbb{C}(\boldsymbol{s}_i, \boldsymbol{s}_j)\}_{ij}$ is the $N \times N$ covariance matrix governed by $\mathbb{C}(\boldsymbol{s}_i, \boldsymbol{s}_j)$ (e.g. the Matérn covariance function). From this definition, the appealing properties of the Gaussian distribution (e.g. Gaussian marginal and conditional distributions) have rendered the GP an indispensable tool for any spatial data analyst to perform such tasks as kriging (spatial prediction) and proper uncertainty quantification.

With the modern onset of larger and larger spatial datasets, however, the use of Gaussian processes for scientific discovery has been hindered by computational intractability. Specifically, evaluating the density in (1) requires $\mathcal{O}(N^3)$ operations and $\mathcal{O}(N^2)$ memory which can quickly overwhelm computing systems when N is only moderately large. Early solutions to this problem included factoring (1) into a series of conditional distributions (Vecchia 1988, Stein et al. 2004), the use of pseudo-likelihoods (Varin et al. 2011, Eidsvik et al. 2014), modeling in the spectral domain (Fuentes 2007) or using tapered covariance functions (Furrer et al. 2006, Kaufman et al. 2008, Stein 2013). Beginning in the late 2000's, several approaches based on low rank approximations to Gaussian processes were developed (or became popular) including discrete process convolutions (Higdon 2002, Lemos and Sansó 2009), fixed rank kriging (Cressie and Johannesson 2008, Kang and Cressie 2011, Katzfuss and Cressie 2011), predictive processes (Banerjee et al. 2008, Finley et al. 2009), lattice kriging (Nychka et al. 2015) and stochastic partial differential equations (Lind-

gren et al. 2011). Sun et al. (2012) and Bradley et al. (2016) provide exceptional reviews of these methods and demonstrate their effectiveness for modeling spatial data.

After several years of their use, however, scientists have started to observe shortcomings in many of the above methods for approximating GPs such as the propensity to oversmooth the data (Simpson et al. 2012, Stein 2014) and even, for some of these methods, an upper limit on the size of the dataset that can be modeled. Hence, recent scientific research in this area has focused on the efficient use of modern computing platforms and the development of methods that are parallelizable. For example, Paciorek et al. (2013) show how (1) can be calculated using parallel computing while Katzfuss and Hammerling (2017) and Katzfuss (2017) develop a basis-function approach that lends itself to distributed computing. Alternatively, Barbian and Assunção (2017) and Guhaniyogi and Banerjee (2017) propose dividing the data into a large number of subsets, draw inference on the subsets in parallel and then combining the inferences. Datta et al. (2016a,c) build upon Vecchia (1988) by developing novel approaches to factoring (1) as a series of conditional distributions based only on nearest neighbors.

Given the plethora of choices to analyze large spatially correlated data, for this paper, we seek to not only provide a review of modern methods to analyze massive spatial datasets, but also compare the methods in a unique way. Specifically, this research implements the common task framework of Wikle et al. (2017) by describing the outcome of a friendly competition between various research groups across the globe who each implemented their own method to analyze the same spatial datasets. That is, to compare the various methods, several research groups were provided with two spatial datasets (one simulated and one real) with a portion of each dataset removed to validate predictions (research groups were not provided with the removed portion so that this study is "blinded"). Each group then implemented their unique method and provided a prediction (and prediction interval or standard error) of the spatial process at the held out locations. The predictions were compared by a third party and are summarized herein.

The competition described herein is unique and novel in that, typically, comparisons/reviews

of various methods is done by a single research group implementing each method. However, single research groups may be more or less acquainted with some methods leading to a possibly unfair comparison with those methods they are less familiar with. In contrast, for the comparison/competition here, each method was implemented by a research group with strong expertise in the method and who is well-versed in any possible intricacies associated with its use. Hence, in terms of scientific contributions, this paper (i) serves as a valuable review and (ii) comparison of spatial methods for large datasets, (iii) provides code to implement each method to practitioners (see supplementary materials) and (iv) establishes a framework for future studies to follow when comparing various analytical methods.

The remainder of this paper is organized as follows. Section 2 gives a brief background on each method. Section 3 provides the setting for the comparison along with background on the datasets. Section 4 then summarizes the results of the comparison in terms of predictive accuracy, uncertainty quantification and computation time. Section 5 draws conclusions from this study and highlights future research areas for the analysis of massive spatial data.

2. OVERVIEW OF METHODS FOR ANALYZING LARGE SPATIAL DATA

2.1 Fixed Rank Kriging

Fixed Rank Kriging (FRK, Cressie and Johannesson 2006, 2008) is built around the concept of a *spatial random effects* (SRE) model. In FRK, one models the process $\widetilde{Y}(\boldsymbol{s}), \boldsymbol{s} \in D$, as

$$\widetilde{Y}(\boldsymbol{s}) = \mu(\boldsymbol{s}) + w(\boldsymbol{s}) + \xi(\boldsymbol{s}), \quad \boldsymbol{s} \in D, \tag{2}$$

where $\mu(\boldsymbol{s})$ is the mean function that is itself modeled as a linear combination of known covariates (i.e. $\mu(\boldsymbol{s}) = \boldsymbol{X}'(\boldsymbol{s})\boldsymbol{\beta}$ as above), $w(\boldsymbol{s})$ is the SRE model, and $\xi(\boldsymbol{s})$ is a fine-scale process, modelled to be spatially uncorrelated with variance σ^2_ξ. The process $\xi(\boldsymbol{s})$ in (5) is designed to soak up

variability in $\widetilde{Y}(\boldsymbol{s})$ not accounted for by $w(\boldsymbol{s})$.

The primary assumption of FRK is that the spatial process $w(\cdot)$ can be decomposed into a linear combination of K basis functions $\boldsymbol{h}(\boldsymbol{s}) = (h_1(\boldsymbol{s}), \ldots, h_k(\boldsymbol{s}))'$, $\boldsymbol{s} \in D$, and K basis function coefficients $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_K)'$ such that,

$$w(\boldsymbol{s}) = \sum_{k=1}^{K} h_k(\boldsymbol{s})\theta_k, \quad \boldsymbol{s} \in \mathcal{D}. \tag{3}$$

The use of K basis functions ensures that all estimation and prediction equations only contain inverses of matrices of size $K \times K$, where $K \ll N$. In practice, the set $\{h_k(\cdot)\}$ in (3) is comprised of functions at R different resolutions such that (3) can also be written as

$$w(\boldsymbol{s}) = \sum_{r=1}^{R} \sum_{k=1}^{K_r} h_{rk}(\boldsymbol{s})\theta_{rk}, \quad \boldsymbol{s} \in D, \tag{4}$$

where $h_{rk}(\boldsymbol{s})$ is the k^{th} spatial basis function at the r^{th} resolution with associated coefficient θ_{rk}, and K_r is the number of basis functions at the r^{th} resolution, such that $K = \sum_{r=1}^{R} K_r$ is the total number of basis functions used. For this research, we used $R = 3$ resolutions of bisquare basis functions following Cressie and Johannesson (2008).

The coefficients $\boldsymbol{\theta} = \{\theta_{rk}\}$ have $\mathbb{V}\text{ar}(\boldsymbol{\theta}) = \boldsymbol{S}(\boldsymbol{\phi})$ with covariance parameters $\boldsymbol{\phi}$ that need to be estimated. In this work, $\boldsymbol{S}(\boldsymbol{\phi})$ is a block-diagonal matrix composed from R dense matrices, where the r^{th} block has i, jth element $\exp(-d_r(i,j)/\phi_r)$ and $d_r(i,j)$ is the distance between the centroids of the i^{th} and j^{th} basis function at the r^{th} resolution, and $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_R)'$ are the spatial correlation parameters of the exponential correlation function. Note that $\boldsymbol{S}(\boldsymbol{\phi})$ can also be unstructured in which case $K(K+1)/2$ parameters need to be estimated, however this case is not considered here.

There are several variants of FRK. In this work, we use the implementation by Zammit-Mangion and Cressie (2017) which comes in the form of the `R` package `FRK`, available from the

Comprehensive R Archive Network (CRAN). In this paper we utlize v0.1.6 of that package. In FRK the model for $\widetilde{Y}(\boldsymbol{s}), \boldsymbol{s} \in D$, is composed as in (2). FRK further assumes that recorded observations $Y(\boldsymbol{s}_i)$ are noisy readings of $\widetilde{Y}(\boldsymbol{s}_i), i = 1, \ldots, N$, such that

$$Y(\boldsymbol{s}_i) = \widetilde{Y}(\boldsymbol{s}_i) + \varepsilon(\boldsymbol{s}_i), \quad i = 1, \ldots, N, \tag{5}$$

where for $i = 1, \ldots, N$, $\varepsilon(\boldsymbol{s}_i)$ denotes independent and identically normally distributed measurement error with mean 0 and known measurement error variance σ^2_ε. More details on the implementation of FRK for this study are included in the supplementary materials.

2.2 LatticeKrig

LatticeKrig (LK, Nychka et al. 2015) uses nearly the same setup as is employed by FRK. Specifically, LK assumes the model (2) and (5) but omits the fine-scale process $\xi(\cdot)$. Further, for $w(\boldsymbol{s})$, LK follows the multiresolution approach in (4), but LK uses a different structure and constraints than FRK. First, the marginal variance of each resolution $\boldsymbol{h}'_r(\boldsymbol{s})\boldsymbol{\theta}_r$ where $\boldsymbol{h}'_r(\boldsymbol{s}) = (h_{r1}(\boldsymbol{s}), \ldots, h_{rK_r}(\boldsymbol{s}))'$ are the basis functions of the r^{th} resolution with coefficients $\boldsymbol{\theta}_r = (\theta_{r1}, \ldots, \theta_{rK_r})'$ is constrained to be $\sigma^2_w \alpha_r$ where $\sigma^2_w, \alpha_r > 0$ and $\sum_{r=1}^R \alpha_r = 1$. To further reduce the number of parameters, LK sets $\alpha_r \sim r^{-\nu}$ where ν is a single free parameter.

LatticeKrig obtains multiresolution radial basis functions by translating and scaling a radial function in the following manner. Let $\boldsymbol{u}_{rk}$ for $r = 1, \ldots, R$ and $k = 1, \ldots, K_r$ denote a regular grid of K_r points on $\mathcal{D}$ corresponding to resolution r. For this article, LK defines

$$h_{rk}(\boldsymbol{s}) = \psi(\|\boldsymbol{s} - \boldsymbol{u}_{rk}\|/\theta_r)$$

where the distance is taken to be Euclidean because the spatial region in this case is of small

geographic extent and $\theta_r = 2^{-r}$. Further, LK defines

$$\psi(d) \propto \begin{cases} \frac{1}{3}(1-d(\boldsymbol{s}))^6(35d^2+18d+3) & \text{if } d \leq 1 \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

which are Wendland polynomials and are positive definite (an attractive property when the basis is used for interpolation). Finally, the basis functions in (6) are normalized at each resolution so that the process marginal variance at all $\boldsymbol{s}$ is $\sigma_w^2\alpha_r$. This reduces edge effects and makes for a better approximation to a stationary covariance function.

LatticeKrig assumes the coefficients at each resolution $\boldsymbol{\theta}_r = (\theta_{r1}, \ldots, \theta_{rK_r})'$ are independent (similar to the block diagonal structure used in FRK) and follow a multivariate normal distribution with covariance $\boldsymbol{Q}_r^{-1}$ parameterized by a single parameter ϕ_r. Because the locations $\{\boldsymbol{u}_{rk}\}_{k=1}^{K_r}$ are prescribed to be a regular grid, LK uses a spatial autoregression/Markov random field (see Banerjee et al. 2014, Section 4.4) structure for $\boldsymbol{Q}_r^{-1}$ leading to sparsity and computational tractability. Furthermore, because $\boldsymbol{Q}_r$ is sparse, LK can set K to be very large (as in this competition greater than N) without much additional computational cost.

2.3 Predictive Processes

For the predictive process (PP) approach, let $\boldsymbol{s}_1^\star, \ldots, \boldsymbol{s}_K^\star$ denote a set of "knot" locations well dispersed over the spatial domain $\mathcal{D}$. Assume that the SREs ($w(\boldsymbol{s})$) in (2) follow a mean zero Gaussian process with covariance function $\mathbb{C}(\boldsymbol{s}, \boldsymbol{s}') = \sigma_w^2\rho(\boldsymbol{s}, \boldsymbol{s}')$ where $\rho(\cdot,\cdot)$ is a postive definite correlation function. Under this Gaussian process assumption, the SREs $\boldsymbol{w}^\star = (w(\boldsymbol{s}_1^\star), \ldots, w(\boldsymbol{s}_K^\star))' \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{w^\star})$ where $\boldsymbol{\Sigma}_{w^\star}$ is a $K \times K$ covariance matrix with ij^{th} element $\mathbb{C}(\boldsymbol{s}_i^\star, \boldsymbol{s}_j^\star)$. The PP approach exploits the Gaussian process assumption for the SREs and replaces $w(\boldsymbol{s})$ in (2) with

$$\widetilde{w}(\boldsymbol{s}) = \mathbb{C}'(\boldsymbol{s}, \boldsymbol{s}^\star)\boldsymbol{\Sigma}_{w^\star}^{-1}\boldsymbol{w}^\star \tag{7}$$

where $\mathbb{C}(\boldsymbol{s}, \boldsymbol{s}^\star) = (\mathbb{C}(\boldsymbol{s}, \boldsymbol{s}_1^\star), \ldots, \mathbb{C}(\boldsymbol{s}, \boldsymbol{s}_K^\star))'$. Note that (7) can be equivalently written as the basis function expression given above in (3) where the basis functions are $\mathbb{C}(\boldsymbol{s}, \boldsymbol{s}^\star)\boldsymbol{\Sigma}_{w^\star}^{-1}$ and $\boldsymbol{w}^\star$ effectively plays the roll of the basis coefficients.

Finley et al. (2009) noted that the basis function expansion in (7) systematically underestimates the marginal variance σ_w^2 from the original process. That is, $\mathbb{V}\text{ar}(\widetilde{w}(\boldsymbol{s})) = \mathbb{C}'(\boldsymbol{s}, \boldsymbol{s}^\star)\boldsymbol{\Sigma}_{w^\star}^{-1}\mathbb{C}'(\boldsymbol{s}, \boldsymbol{s}^\star) \leq \sigma_w^2$. To counterbalance this underestimation of the variance, Finley et al. (2009) use the structure in (5),

$$Y(\boldsymbol{s}) = \mu(\boldsymbol{s}) + \widetilde{w}(\boldsymbol{s}) + \xi(\boldsymbol{s}) + \varepsilon(\boldsymbol{s}) \tag{8}$$

where $\xi(\boldsymbol{s})$ are spatially independent with distribution $\mathcal{N}(0, \sigma_w^2 - \mathbb{C}'(\boldsymbol{s}, \boldsymbol{s}^\star)\boldsymbol{\Sigma}_{w^\star}^{-1}\mathbb{C}'(\boldsymbol{s}, \boldsymbol{s}^\star))$ such that $\mathbb{V}\text{ar}(\widetilde{w}(\boldsymbol{s}) + \xi(\boldsymbol{s})) = \sigma_w^2$ as in the original parent process.

As with FRK and LatticeKrig, the associated likelihood under (8) only requires calculating the inverse and determinant of a dense $K \times K$ matrix and diagonal $N \times N$ matrices which results in massive computational savings when $K \ll N$ and K is small. However, one advertised advantage of using the PP approach as opposed to FRK or LatticeKrig is that the PP basis functions are completely determined by the choice of covariance function $\mathbb{C}(\cdot, \cdot)$. Hence, the PP approach is unaltered even when considering modeling complexities such as anisotropy, non-stationarity or even multivariate processes. At the same time, however, when $\mathbb{C}(\cdot, \cdot)$ is governed by unknown parameters (which is nearly always the case) the PP basis functions need to be calculated iteratively rather than once as in FRK or LatticeKrig which will subsequently increase computation time.

2.4 Spatial Partitioning

Let the spatial domain $\mathcal{D} = \bigcup_{d=1}^{D} \mathcal{D}_d$ where $\mathcal{D}_1, \ldots, \mathcal{D}_D$ are subregions that form a partition (i.e. $\mathcal{D}_{d_1} \bigcap \mathcal{D}_{d_2} = \emptyset$ for all $d_1 \neq d_2$). The modeling approach based on spatial partitioning is to assume *conditional* dependence between observations within a subregion and *conditional* inde-

pendence between observations across subregions. More specifically, if $\boldsymbol{Y}_d = \{Y(\boldsymbol{s}_i) : \boldsymbol{s}_i \in \mathcal{D}_d\}$ where $d = 1, \ldots, D$, then

$$\boldsymbol{Y}_d \overset{ind}{\sim} \mathcal{N}\left(\boldsymbol{X}_d\boldsymbol{\beta} + \boldsymbol{H}_d\boldsymbol{\theta}, \boldsymbol{\Sigma}(\boldsymbol{\phi}_d)\right) \tag{9}$$

where $\boldsymbol{X}_d$ is a design matrix containing covariates associated with $\boldsymbol{Y}_d$, $\boldsymbol{H}_d$ is a matrix of spatial basis functions (such as those used in predictive processes, fixed rank kriging or lattice kriging mentioned above) and $\boldsymbol{\Sigma}(\boldsymbol{\phi}_d)$ is the covariance matrix for subregion d governed by covariance parameters $\boldsymbol{\phi}_d$ (e.g. decay and smoothness parameters). Notice that, in (9) each subregion shares common $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ parameters which allows smoothing across subregions (hence, $Y_{d_1} \perp\!\!\!\perp Y_{d_2}$ for $d_1 \neq d_2$ conditional on the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$). Further, the assumption of independence across subregions allows the likelihood for $\boldsymbol{\beta}, \boldsymbol{\theta}$ and $\boldsymbol{\phi}_d$ is to be computed in parallel thereby facilitating computation.

By way of distinction, this approach is inherently different from the "divide and conquer" approach (Liang et al. 2013, Barbian and Assunção 2017). In the divide and conquer approach, the full dataset is subsampled, the model is fit to each subset and the results across subsamples are pooled. In contrast, the spatial partition approach uses all the data simultaneously in obtaining estimates, but the independence across regions facilitates computation.

The key to implementing the spatial partitioning approach is the choice of partition and the literature is replete with various options. *A priori* methods to define the spatial partitioning include partitioning the region into equal areas (Sang et al. 2011), partitioning based on centroid clustering (Knorr-Held and Raßer 2000, Kim et al. 2005) and hierarchical clustering based on spatial gradients (Anderson et al. 2014, Heaton et al. 2017). Alternatively, model-based approaches to spatial partitioning include treed regression (Konomi et al. 2014) and mixture modeling (Neelon et al. 2014) but these approaches typically require more computation. For this analysis, a couple of different partitioning schemes were considered, but each scheme resulted in approximately

equivalent model fit to the training data. Hence, based on the results from the training data, for the competition below we used an equal area partition of approximately 6000 observations per subregion.

2.5 Covariance Tapering

The idea of covariance tapering is based on the fact that many entries in the covariance matrix $\boldsymbol{\Sigma}$ in (1) are close to zero and associated location pairs could be considered as essentially independent. Covariance tapering multiplies the covariance function $\mathbb{C}(\boldsymbol{s}_i, \boldsymbol{s}_j)$ with a compactly supported covariance function, resulting in another positive definite covariance function but with compact support. From a theoretical perspective, covariance tapering (in the framework of infill-asymptotics) is using the concept of Gaussian equivalent measures and mis-specified covariance functions (see, e.g., Stein 1999 and references therein). Subsequently, Furrer et al. (2006) have assumed a second-order stationary and isotropic Matérn covariance to show asymptotic optimality for prediction under tapering. This idea has been extended to different covariance structures (Stein 2013), non-Gaussian response (Hirano and Yajima 2013) and multivariate and/or spatio-temporal setting (Furrer et al. 2016).

From a computational aspect, the compact support of the resulting covariance function provides the computational savings needed by employing sparse matrix algorithms to efficiently solve systems of linear equations. More precisely, to evaluate density (1), a Cholesky factorization for $\boldsymbol{\Sigma}$ is performed followed by two solves of triangular systems. For typical spatial data settings, the solve algorithm is effectively linear in the number of observations.

In practice, one- and two-taper approaches exist (see Kaufman et al. 2008, Du et al. 2009, Wang and Loh 2011, Bevilacqua et al. 2016, for relevant literature). To distinguish the two approaches,

notice that the likelihood in (1) can be rewritten as

$$f_{\boldsymbol{Y}}(\boldsymbol{y}) = \left(\frac{1}{\sqrt{2\pi}}\right)^{N} |\boldsymbol{\Sigma}|^{-1/2} \text{etr}\left\{-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{\mu})(\boldsymbol{y}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}\right\} \tag{10}$$

where $\text{etr}(\boldsymbol{A}) = \exp(\text{trace}(A))$. In the one-taper setting, only the covariance is tapered such that $\boldsymbol{\Sigma}$ in (10) is replaced by $\boldsymbol{\Sigma} \odot \boldsymbol{T}$ where "$\odot$" denotes the Hadamard product and $\boldsymbol{T}$ is the $N \times N$ tapering matrix. In the two-tapered approach both the covariance and empirical covariance are affected such that not only is $\boldsymbol{\Sigma}$ replaced by $\boldsymbol{\Sigma} \odot \boldsymbol{T}$ but $(\boldsymbol{y}-\boldsymbol{\mu})(\boldsymbol{y}-\boldsymbol{\mu})'$ is replaced by $(\boldsymbol{y}-\boldsymbol{\mu})(\boldsymbol{y}-\boldsymbol{\mu})' \odot \boldsymbol{T}$. The one-taper equation results in biased estimates of model parameters while the two-taper approach is based on estimating equations (and is, therefore, unbiased) but comes at the price of a severe loss of computational efficiency. If the one-taper biased estimates of model parameters are used for prediction, the biases may result in some loss of predictive accuracy (Furrer et al. 2016).

Although tapering can be adapted to better take into account uneven densities of locations and complex anisotropies, we use a simple straight-forward approach for this competition. The implementation here relies almost exclusively on the `R` package `spam` (Furrer and Sain 2010, Furrer 2016). In view of numerical stability, we have decided to use a method-of-moment type estimation procedure compared to a one-taper likelihood approach.

2.6 Multiresolution Approximations

The multi-resolution approximation (MRA) can be viewed as a combination of several previously described approaches. Similar to FRK or LatticeKrig, the MRA expresses the spatial process of interest $w(\boldsymbol{s})$ in (2) as a weighted sum of *compactly* supported basis functions at different resolutions as in (4). In contrast to FRK or LatticeKrig, the MRA basis functions and the prior distribution of the corresponding weights are chosen using the predictive-process approach to automatically adapt to any given covariance function $\mathbb{C}(\cdot)$, and so the MRA can adjust flexibly to a desired spatial smoothness and dependence structure. Scalability of the MRA is ensured in that for

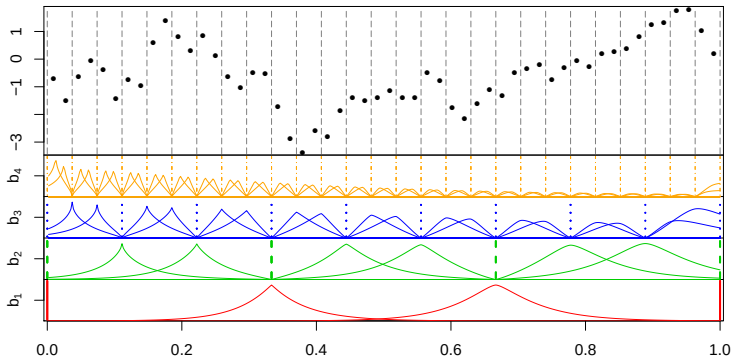

Figure 1. A toy example of simulated observations (black dots) with a covariance function $\mathbb{C}$ with increasing smoothness on a one-dimensional spatial domain $\mathcal{D} = [0, 1]$, together with a multi-resolution approximation (MRA) with $M = 4$ resolutions with 3 subregions per region (vertical lines) and $r_0 = 2$ basis functions per region. The basis functions and their weights (symbolized by the height of the functions) adjust to the changing smoothness, here increasing from left to right.

increasing resolution, the number of basis functions increases while the support of each function (i.e., the part of the spatial domain in which it is nonzero) decreases. Decreasing support (and increasing sparsity of the covariance matrices of the corresponding weights) is achieved either by increasingly severe tapering of the covariance function (MRA-taper) or by recursively partitioning the spatial domain (MRA-block; Katzfuss 2017). This can lead to (nearly) exact approximations with quasilinear computational complexity.

While the MRA-taper has some attractive smoothness properties, we focus here on the MRA-block which is based on a recursive partitioning of the domain $\mathcal{D}$ into smaller and smaller subregions up to some level M. Within each (sub-)region at each resolution, there is a small number, say r_0, of basis functions. The resulting approximation of the process (including its variance and smoothness) in each region at resolution M is exact. In addition, it is feasible to compute and store the joint posterior covariance matrix (i.e., not just its inverse as with related approaches) for a large number of prediction locations as a product of two sparse matrices. Figure 1 illustrates the MRA basis functions in a toy example.

The MRA-block is designed to take full advantage of high-performance computing systems, in that inference is well suited for massively distributed computing, with limited communication over-

head. The computational task is split into small parts by assigning a computational node to each region of the recursive partitioning. The nodes then deal in parallel with the basis functions corresponding to their assigned regions. This can lead to polylogarithmic computational complexity, and has enabled successful application of the MRA to datasets with over 100 million observations. For this project, we use $M = 9$ levels, partition each domain in 2 parts and set the number of basis function in each partition to $r_0 = 64$. Given the comparatively small data size, the code is executed sequentially, which proved faster for this competition.

2.7 Nearest Neighbor Processes

The nearest neighbor Gaussian process (NNGP) developed in Datta et al. (2016a) and Datta et al. (2016b) is defined from the conditional specification of the joint distribution of the SREs in (2). Let $w(\boldsymbol{s})$ in (2) follow a mean zero Gaussian process with $\mathbb{C}(\boldsymbol{s}, \boldsymbol{s}') = \sigma_w^2 \rho(\boldsymbol{s}, \boldsymbol{s}')$ where $\rho(\cdot)$ is a positive definite correlation function. Factoring the joint density of $w(\boldsymbol{s}_1), \ldots, w(\boldsymbol{s}_N)$ into a series of conditional distributions yields that $w(\boldsymbol{s}_1) = 0 + \eta(\boldsymbol{s}_1)$ and

$$w(\boldsymbol{s}_i) \mid \boldsymbol{w}_{1:(i-1)} = \mathbb{C}'(\boldsymbol{s}_1, \boldsymbol{s}_{1:(i-1)}) \boldsymbol{\Sigma}_{1:(i-1)}^{-1} \boldsymbol{w}_{1:(i-1)} + \eta(\boldsymbol{s}_i) \tag{11}$$

where $\boldsymbol{w}_{1:(i-1)} = (w(\boldsymbol{s}_1), \ldots, w(\boldsymbol{s}_{i-1}))'$, $\mathbb{C}(\boldsymbol{s}_1, \boldsymbol{s}_{1:(i-1)}) = (\mathbb{C}(\boldsymbol{s}_i, \boldsymbol{s}_1), \ldots, \mathbb{C}(\boldsymbol{s}_i, \boldsymbol{s}_{i-1})'$, $\boldsymbol{\Sigma}_{1:(i-1)} = \mathbb{V}\text{ar}(\boldsymbol{w}_{1:(i-1)})$ and η's are independent, mean zero, normally distributed random variables. More compactly, (11) is equivalent to $\boldsymbol{w} = \boldsymbol{A}\boldsymbol{w} + \boldsymbol{\eta}$ where $\boldsymbol{A} = (a_{ij})$ is a lower triangular matrix with zeroes along the diagonal and $\boldsymbol{\eta} = (\eta(\boldsymbol{s}_1), \ldots, \eta(\boldsymbol{s}_n))' \sim N(0, \boldsymbol{D})$ with diagonal entries $\mathbb{C}(\boldsymbol{s}_i, \boldsymbol{s}_i) - \mathbb{C}'(\boldsymbol{s}_1, \boldsymbol{s}_{1:(i-1)}) \boldsymbol{\Sigma}_{1:(i-1)}^{-1} \mathbb{C}(\boldsymbol{s}_1, \boldsymbol{s}_{1:(i-1)})$. This effectuates a joint distribution $\boldsymbol{w} \sim N(0, \boldsymbol{\Sigma})$ where $\boldsymbol{\Sigma}^{-1} = (\boldsymbol{I} - \boldsymbol{A})' \boldsymbol{D}^{-1} (\boldsymbol{I} - \boldsymbol{A})$. Furthermore, when predicting for any $\boldsymbol{s} \notin \{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N\}$, one can define

$$w(\boldsymbol{s}) \mid \boldsymbol{w}_{1:N} = \boldsymbol{a}'(s) \boldsymbol{w}_{1:N} + \eta(\boldsymbol{s}) \tag{12}$$

similar to (11).

A sparse formulation of $\boldsymbol{A}$ ensures that evaluating the likelihood of $\boldsymbol{w}$ (and, hence, of $\boldsymbol{Y}$) will be computationally scalable. Because spatial covariances decrease with increasing distance, Vecchia (1988) demonstrated that replacing the conditional set $\boldsymbol{w}_{1:(i-1)}$ by the smaller set of m nearest neighbors (in terms of Euclidean distance) of $\boldsymbol{s}_i$ provides an excellent approximation to the conditional density in (11). Datta et al. (2016a) demonstrated that this is equivalent to $\boldsymbol{A}$ having at-most m non-zero entries in each row and thereby corresponds to a proper probability distribution. Similarly, for prediction at a new location $\boldsymbol{s}$, a sparse $\boldsymbol{a}(\boldsymbol{s})$ in (12) is constructed based on m-nearest neighbors of $\boldsymbol{s}$ among $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N$. The resulting Gaussian Process is referred to as the Nearest Neighbor Gaussian Process (NNGP) and computation primarily involves small $m \times m$ matrix operations. Generalizing the use of nearest neighbors from expedient likelihood evaluations as in Vecchia (1988) and Stein et al. (2004) to the well defined NNGP on the entire domain enables fully Bayesian inference and coherent recovery of the latent SREs.

Using an NNGP prior for $Y(\boldsymbol{s}) - \boldsymbol{X}'(\boldsymbol{s})\boldsymbol{\beta}$, the model can be written as $\boldsymbol{Y} \sim N(\boldsymbol{X}\boldsymbol{\beta}, \widetilde{\boldsymbol{\Sigma}}(\boldsymbol{\phi}))$ where $\widetilde{\boldsymbol{\Sigma}}$ is the NNGP covariance matrix derived from the full GP. A Bayesian specification is completed by specifying priors for the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$. For this application, the covariance function $\mathbb{C}$ consists of an stationary exponential GP with variance σ^2 and range ϕ and a nugget process with variance σ^2_ε (see (5)). We assign a normal prior for $\boldsymbol{\beta}$, inverse gamma priors for σ^2_w and σ^2_ε and a uniform prior for ϕ. A Gibbs sampler for the model involves conjugate updates for $\boldsymbol{\beta}$ and metropolis random walk updates for $\boldsymbol{\phi} = (\sigma^2_w, \sigma^2_\varepsilon, \phi)'$.

Letting $\alpha = \sigma^2_\varepsilon/\sigma^2_w$, the model can also be expressed as $\boldsymbol{Y} \sim N(\boldsymbol{X}\boldsymbol{\beta}, \sigma^2_w\widetilde{\boldsymbol{R}}(\phi, \alpha))$ where $\widetilde{\boldsymbol{R}}$ is the NNGP matrix derived from $\boldsymbol{C}(\phi) + \alpha\boldsymbol{I}$, $\boldsymbol{C}(\phi)$ being the correlation matrix of the exponential GP. Fixing α and ϕ gives a conjugate Normal-Inverse Gamma posterior distribution for $\boldsymbol{\beta}$ and σ^2_w. Predictive distributions for $y(s)$ at new locations can also be obtained as t-distributions. The fixed values of α and ϕ can be chosen from a grid-search by minimizing root mean square predictive error score based on K-fold cross validation. This hybrid approach departs from fully Bayesian philosophy by using hyper-parameter tuning. However, it offers a pragmatic solution for massive

spatial datasets. We refer to this model as the *conjugate NNGP* model and the fully Bayesian approach described above as the *response NNGP* model. Detailed algorithms for both the models are provided in Finley et al. (2017b). NNGP models for analyzing massive spatial data are available on CRAN as the R-package *spNNGP* (Finley et al. 2017a).

2.8 Stochastic PDEs

The stochastic partial differential equation approach (SPDE) is based on the equivalence between Matérn covariance fields and stochastic PDEs, in combination with the Markov property that on 2-dimensional domains holds for integer valued smoothness parameters in the Matérn family. The starting point is a basis expansion for $w(\boldsymbol{s})$ of the form (3), where the basis functions $h_k(\boldsymbol{s})$ are chosen to be piecewise linear on a triangulation of the domain (Lindgren et al. 2011). The optimal joint distribution for the θ_k coefficients is obtained through a finite element construction, which leads to a sparse inverse covariance matrix (precision) $\boldsymbol{Q}_\theta(\boldsymbol{\phi})$. The precision matrix elements are polynomials in the precision and inverse range parameters ($1/\phi_\sigma^2$ and $1/\phi_r$), with sparse matrix coefficients that are determined solely by the choice of triangulation. This differs from the sequential Markov construction of the NNGP method which instead constructs a square-root free $\boldsymbol{LDL'}$ Cholesky decomposition of its resulting precision matrix (in a reverse order permutation of the elements).

The spatial process is specified through a joint Gaussian model for $\boldsymbol{z} = (\boldsymbol{\theta}, \boldsymbol{\beta})$ with prior mean $\mathbf{0}$ and block-diagonal precision $\boldsymbol{Q}_z = \text{diag}(\boldsymbol{Q}_\theta, \boldsymbol{Q}_\beta)$, where $\boldsymbol{Q}_\beta = \boldsymbol{I} \cdot 10^{-8}$ gives a vague prior for $\boldsymbol{\beta}$. Introducing the sparse basis evaluation matrix $\boldsymbol{H}$ with elements $H_{ij} = h_j(\boldsymbol{s}_i)$ and covariate matrix $\boldsymbol{X} = X_j(\boldsymbol{s}_i)$, the observation model is then $\boldsymbol{Y} = \boldsymbol{X\beta} + \boldsymbol{H\theta} + \boldsymbol{\varepsilon}$. The design matrix for the joint vector $\boldsymbol{z}$ is $\boldsymbol{A} = (\boldsymbol{H}, \boldsymbol{X})$, and $\boldsymbol{\varepsilon}$ is a zero mean observation noise vector with diagonal precision $\boldsymbol{Q}_\varepsilon = \boldsymbol{I}/\sigma_\varepsilon^2$.

Using the precision based equations for multivariate Normal distributions, the conditional precision and expectation for $\boldsymbol{z}$ are given by $\boldsymbol{Q}_{z|y} = \boldsymbol{Q}_z + \boldsymbol{A'Q}_\varepsilon\boldsymbol{A}$ and $\boldsymbol{\mu}_{z|y} = \boldsymbol{Q}_{z|y}^{-1}\boldsymbol{A'Q}_\varepsilon\boldsymbol{Y}$, where

sparse Cholesky factorisation of $\boldsymbol{Q}_{z|y}$ is used for the linear solve. The elements of $\boldsymbol{z}$ are automatically reordered to keep the Cholesky factors as sparse as possible. The resulting computational and storage cost for the posterior predictions and multivariate Gaussian likelihood of a spatial Gaussian Markov random field of this type with K basis functions is $\mathcal{O}(K^{3/2})$. Since the direct solver does not take advantage of the stationarity of the model, the same prediction cost would apply to non-stationary models. For larger problems, more easily parallelizeable iterative sparse solvers (e.g. multigrid) can be applied, but for the relatively small size of the problem here, the straightforward implementation of a direct solver is likely preferable. The posterior covariance elements of $\boldsymbol{Q}_{z|y}^{-1}$ corresponding to the non-zero structure of $\boldsymbol{Q}_{z|y}$ are obtained through Takahashi recursions as a post-processing step on the Cholesky factor of $\boldsymbol{Q}_{z|y}$ (see Rue et al. 2009). These elements are precisely the ones needed to compute the final predictive variances $\mathbb{V}\text{ar}[\mu(\boldsymbol{s}_0) + w(\boldsymbol{s}_0) + \varepsilon_0 \mid \boldsymbol{Y}]$ for each prediction location $\boldsymbol{s}_0$.

The triangulation nodes were here chosen to coincide with the observation lattice, and in order to avoid unwanted boundary effects, the triangulation extends a short distance outside the domain. This extension has only a small effect on the computational cost, since the triangles are allowed to be larger than inside the domain of interest, and therefore the extension doesn't need as many nodes as in a regular lattice extension. In addition, the exponential covariance is a Matérn covariance with smoothness 0.5, and hence is not Markovian on $\mathbb{R}^2$. Where the LK method approaches this by using a sum of several Markovian components, the SPDE implementation in `INLA` (Rue et al. 2017) instead uses a parsimonious Markovian spectral approximation for a single field. The resulting model is a second order Markov random field on the coefficients $\{\theta_k\}$. For details of the approximation see the authors' response to the discussion of Lindgren et al. (2011).

The implementation of the SPDE method used here is based on the `R` package `INLA` (Rue et al. 2017), which is aimed at Bayesian inference for latent Gaussian models (in particular Bayesian generalised linear, additive, and mixed models) using integrated nested Laplace approximations (Rue et al. 2009). The package supports significantly more general models than considered here,

in particular non-Gaussian observation models that lead to non-Guassian posterior distributions for the random field components. The parameter optimisation for $\boldsymbol{\phi} = (\phi_r, \phi_\sigma, \sigma_\varepsilon^2)$ uses general numerical log-likelihood derivatives and thus does not take advantage of the special parameter structure that would allow explicit derivative implementations for this simple fully Gaussian model class. In order to avoid some of the unnecessary computations, the full Bayesian inference was therefore turned off, leading to an empirical Bayes estimate of the covariance parameters. Most of the running time is still spent on parameter optimisation, but using the same parameter estimation technique as for LK, in combination with a purely Gaussian implementation, should be expected to significantly reduce the total running time even without specialised code for the derivatives.

2.9 Metakriging

Spatial meta kriging is an approximate Bayesian method that is not tied to any specific model and is partly algorithmic in nature. In particular, any spatial model described above can be used to draw inference from subsets (as described below). From (1), let the $N \times N$ covariance matrix be determined by a set of covariance parameters $\boldsymbol{\phi}$ such that $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\boldsymbol{\phi})$ (e.g. $\boldsymbol{\phi}$ could represent decay parameters from the Matérn covariance function) and $\mu(\boldsymbol{s}) = \boldsymbol{X}'(\boldsymbol{s})\boldsymbol{\beta}$ where $\boldsymbol{X}(\boldsymbol{s})$ is a set of known covariates with unknown coefficients $\boldsymbol{\beta}$. Further, let the sampled locations $\mathcal{S} = \{\boldsymbol{s}_1, ..., \boldsymbol{s}_N\}$ be partitioned into sets $\{\mathcal{S}_1, ..., \mathcal{S}_K\}$ such that $\mathcal{S}_i \cap \mathcal{S}_j = \emptyset$ for $i \neq j$ and the corresponding partition of the data be given by $\{\boldsymbol{y}_k, \boldsymbol{X}_k\}$, for $k = 1, 2, \ldots, K$, where each $\boldsymbol{y}_k$ is $n_k \times 1$ and $\boldsymbol{X}_k$ is $n_k \times p$. Assume that we are able to obtain posterior samples for $\boldsymbol{\Omega} = \{\boldsymbol{\beta}, \boldsymbol{\phi}\}$ from (1) applied independently to each of K subsets of the data in *parallel on different cores*. To be specific, assume that $\boldsymbol{\Omega}_k = \{\boldsymbol{\Omega}_k^{(1)}, \boldsymbol{\Omega}_k^{(2)}, \ldots, \boldsymbol{\Omega}_k^{(M)}\}$ is a collection of M posterior samples from $p(\boldsymbol{\Omega} \mid \boldsymbol{y}_k)$. We refer to each $p(\boldsymbol{\Omega} \mid \boldsymbol{y}_k)$ as a "subset posterior." The meta-kriging approach we outline below attempts to combine, optimally and meaningfully, these subset posteriors to arrive at a legitimate probability density. We refer to this as the "meta-posterior".

Metakriging relies upon the unique geometric median (GM) of the subset posteriors (Minsker

et al. 2014, Minsker 2015). We envision the individual posterior densities $p_k \equiv p(\boldsymbol{\Omega} \mid \boldsymbol{y}_k)$ to be residing on a Banach space $\mathcal{H}$ equipped with norm $\|\cdot\|_\rho$. The GM is defined as

$$\pi^*(\boldsymbol{\Omega} \mid \boldsymbol{y}) = \arg\min_{\pi \in \mathcal{H}} \sum_{k=1}^{K} \|p_k - \pi\|_\rho \ , \tag{13}$$

where $\boldsymbol{y} = (\boldsymbol{y}_1', \boldsymbol{y}_2', \ldots, \boldsymbol{y}_K')'$. The norm quantifies the distance between any two posterior densities $\pi_1(\cdot)$ and $\pi_2(\cdot)$ as $\|\pi_1 - \pi_2\|_\rho = \| \int \rho(\boldsymbol{\Omega}, \cdot) d(\pi_1 - \pi_2)(\boldsymbol{\Omega})\|$, where $\rho(\cdot)$ is a positive-definite kernel function. In what follows, we assume $\rho(z_1, z_2) = \exp(-||z_1 - z_2||^2)$.

The GM is unique. Further, the geometric median lies in the convex hull of the individual posteriors, so $\pi^*(\boldsymbol{\Omega} \mid \boldsymbol{y})$ is a legitimate probability density. Specifically, $\pi^*(\boldsymbol{\Omega} \mid \boldsymbol{y}) = \sum_{k=1}^{K} \alpha_{\rho,k}(\boldsymbol{y}) p_k$, $\sum_{k=1}^{K} \alpha_{\rho,k}(\boldsymbol{y}) = 1$, each $\alpha_{\rho,k}(\boldsymbol{y})$ being a function of $\rho, \boldsymbol{y}$, so that $\int_{\boldsymbol{\Omega}} \pi^*(\boldsymbol{\Omega} \mid \boldsymbol{y}) d\boldsymbol{\Omega} = 1$.

Computation of the geometric median $\pi^* \equiv \pi^*(\boldsymbol{\Omega} \mid \boldsymbol{y})$ proceeds by employing the popular Weiszfeld's iterative algorithm that estimates $\alpha_{\rho,k}(\boldsymbol{y})$ for every k from the subset posteriors p_k. To further elucidate, we use a well known result that the geometric median π^* satisfies,

$$\pi^* = \frac{\sum_{k=1}^{K} \|p_k - \pi^*\|_\rho^{-1} p_k}{\sum_{k=1}^{K} \|p_k - \pi^*\|_\rho^{-1}}$$

so that $\alpha_{\rho,k}(\boldsymbol{y}) = \|p_k - \pi^*\|_\rho^{-1} / \sum_{j=1}^{K} \|p_k - \pi^*\|_\rho^{-1}$. Since there is no apparent closed form solution for $\alpha_{\rho,k}(\boldsymbol{y})$ that satisfies this equation, one needs to resort to the Weiszfeld iterative algorithm outlined in Minsker et al. (2014) to produce an empirical estimate of $\alpha_{\rho,k}(\boldsymbol{y})$ for all $k = 1, .., K$.

Guhaniyogi and Banerjee (2017) show that, for a large sample, $\pi^*(\cdot \mid \boldsymbol{y})$ provides desirable approximation of the full posterior distribution in certain restrictive settings. It is, therefore, natural to approximate the posterior predictive distribution $p(y(s_0) \mid \boldsymbol{y})$ by the subset posterior predictive distributions $p(y(s_0) \mid \boldsymbol{y}_k)$. Let $\{y(s_0)^{(j,k)}\}_{j=1}^{M}$, $k = 1, \ldots, K$, be samples obtained from the posterior

predictive distribution $p(y(s_0)|\boldsymbol{y}_k)$ from the k-th subset posterior. Then,

$$p(y(s_0) \mid \boldsymbol{y}) \approx \sum_{k=1}^{K} \alpha_{\rho,k}(\boldsymbol{y}) p(y(s_0) \mid \boldsymbol{y}_k) = \sum_{k=1}^{K} \alpha_{\rho,k}(\boldsymbol{y}) \int p(y(s_0) \mid \boldsymbol{\Omega}, \boldsymbol{y}_k) p(\boldsymbol{\Omega} \mid \boldsymbol{y}_k) d\boldsymbol{\Omega} \ ,$$

Therefore, the empirical posterior predictive distribution of the meta posterior is given by $\sum_{k=1}^{K} \sum_{j=1}^{M} \frac{\alpha_{\rho,k}(\boldsymbol{y})}{M} 1_{y(s_0)^{(j,k)}}$, from which the posterior predictive median and the 95% posterior predictive interval for the unobserved $y(s_0)$ are readily available.

One important ingredient of spatial meta kriging (SMK) is partitioning the dataset into subsets. For this article, we adopt a random partitioning scheme that randomly divides data into $K = 30$ exhaustive and mutually exclusive subsets. The random partitioning scheme facilitates each subset to be a reasonable representative of the entire domain, so that each subset posterior acts as a "weak learner" of the full posterior. We have explored more sophisticated partitioning schemes and found similar predictive inference.

For the sake of definiteness, this article uses the Gaussian process model for each subset inference which may lead to higher run time. However, the meta kriging approach lends much more scalability when any of the above models is employed in each subset. In fact, ongoing research in spatial meta kriging includes distributed spatial kriging (DISK) which scales the modified predictive process to millions of observations.

2.10 Gapfill

The gapfill method (Gerber et al. 2016) differs from the other herein presented methods in that it is purely algorithmic, distribution-free, and, in particular, not based on Gaussian processes. Like other prediction methods popular within the satellite imaging community (see Gerber et al. 2016 and Weiss et al. 2014 for reviews), the gapfill method is attractive because of its low computational workload. A key aspect of gapfill is that it is designed for parallel processing, which allows the user to exploit computing resources at different scales including large servers. Parallelization is

enabled by predicting each missing value separately based on only a subset of the data.

To predict the value $Y(\boldsymbol{s}_0)$ at location $\boldsymbol{s}_0$ gapfill first selects a suitable subset $\boldsymbol{A} = \{Y(\boldsymbol{s}_i) : \boldsymbol{s}_i \in \mathcal{N}(\boldsymbol{s}_0)\}$, where $\mathcal{N}(\boldsymbol{s}_0)$ defines a spatial neighborhood around $\boldsymbol{s}_0$. Finding $\boldsymbol{A}$ is formalized with rules, which reassure that $\boldsymbol{A}$ is small but contains enough observed values to inform the prediction. In this study, we require $\boldsymbol{A}$ to have an extent of at least 5×5 pixels and to contain at least 25 non-missing values. Subsequently, the prediction of $Y(\boldsymbol{s}_0)$ is based on $\boldsymbol{A}$ and relies on sorting algorithms and quantile regression. Moreover, prediction intervals are constructed using permutation arguments (see Gerber et al. 2016 for more details on the prediction and uncertainty intervals).

The gapfill method was originally designed for spatio-temporal data, in which case the neighborhood $\mathcal{N}(\boldsymbol{s}_0)$ is defined in terms of the spatial and temporal dimensions of the data. As a consequence, the implementation of gapfill in the R package `gapfill` (Gerber 2017) requires multiple images to work properly. To mimic this situation, we shift the given images by one, two, and three pixels in both directions along the x and y-axes. Then the algorithm is applied to those 13 images in total (one original image and 12 images obtained through shifts of the original image).

2.11 Local Approximate Gaussian Processes

The local approximate Gaussian process (`laGP`, Gramacy and Apley 2015) addresses the big-N problem in GP regression by taking a so-called *transductive* approach to learning, where the fitting scheme is tailored to the prediction problem (Vapnik 1995) as opposed to the usual *inductive* approach of fitting first and predicting later conditional on the fit. A special case of `laGP`, based on nearest neighbors, is simple to describe. In order to predict at $\boldsymbol{s}$, simply train a Gaussian process predictor on the nearest m neighbors to $\boldsymbol{s}$; i.e., use the data subset $\mathcal{Y}_m = \{Y(\boldsymbol{s}_i) : \boldsymbol{s}_i \in \mathcal{N}_m(\boldsymbol{s})\}$, where $\mathcal{N}_m(\boldsymbol{s})$ are the m closest observed locations to $\boldsymbol{s}$ in terms of Euclidean distance. If the data-generating mechanism is not at odds with modeling assumptions (e.g., having a well-specified covariance structure), then one can choose m to be as large as possible, up to computational lim-

itations, in order to obtain an accurate approximation. Observe that this use of nearest neighbors (NNs) for prediction is more akin to the classical statistical/machine learning variety, in contrast to their use in determining the global (inverse) covariance structure as described in Section 2.7.

Interestingly, NNs do not comprise an optimal data subset for prediction under the usual criteria such as mean-squared error. However, finding the best m of $N!/(m!(N-m)!)$ possible choices represents a combinatorially huge search. The `laGP` method generalizes this so-called nearest neighbor prediction algorithm (whose modern form in spatial statistical literature is described by Emery 2009) by approximating that search with a greedy heuristic. First, start with a NN set $\mathcal{Y}_{m_0}(\boldsymbol{s}) = \{Y(\boldsymbol{s}_i) : \boldsymbol{s}_i \in \mathcal{N}_{m_0}(\boldsymbol{s}))$ where $m_0 < m$, and then for $j = m_0 + 1, \ldots, m$ successively choose $\boldsymbol{s}_j$ to augment $\mathcal{Y}_{m_0}$ building up a local design data set one point at a time according to one of several simple objective criteria related to mean-square prediction error. The idea is to repeat in this way until there are m observations in $\mathcal{Y}_m(\boldsymbol{s})$. Gramacy and Apley's preferred variation targets $\boldsymbol{s}_j$ which maximizes the *reduction* in predictive variance at $\boldsymbol{s}$. To recognize a similar *global* design criterion called *active learning Cohn* (Cohn 1996), they dubbed this criterion ALC. Qualitatively, these local ALC designs tend to have a cluster of neighbors and "satellite" points and have been shown to offer demonstrably better predictive properties than NN and even full-data alternatives especially when the data generating mechanism is at odds with the modeling assumptions. The reason is that local fitting offers a way to cope with a certain degree of non-stationarity which is common in many real data settings.

ALC search iterations and GP updating considerations as designs are built up, are carefully engineered to lead to a method whose computations are of $\mathcal{O}(n^3)$ complexity (i.e., the same as the simpler NN alternative). A relatively modest local design size of $m = 50$ typically works well. Moreover, calculations for each $\boldsymbol{s}$ are statistically independent of the next, which means that they can be trivially parallelized. Through a cascade of multi-core, multi-node and GPU parallelization, Gramacy et al. (2014) and Gramacy and Haaland (2016) illustrated how N in the millions, in terms of both training and testing data sizes could be handled (and yield accurate

predictors) with less than an hour of computing time. The `laGP` method has been packaged for R and is available on CRAN (Gramacy 2016). Symmetric multi-core parallelization (via `OpenMP`) and multi-node automations (via the built-in `parallel` package) work out-of-the box. GPU extensions are provided in the source code but require custom compilation.

A disadvantage to local modeling in this fashion is that a global predictive covariance is unavailable. Indeed, the statistically independent nature of calculation is what makes the procedure computationally efficient and parallelizable. In fact, the resulting global predictive surface, over a continuum of predictive $\boldsymbol{s}$-locations, need not even be smooth. However in most visual representations of predictive surfaces it can be difficult to distinguish between a genuinely smooth surface and what is plotted via the `laGP` predictive equations (see Figures 3 and 4 below). Finally, it is worth noting that although `laGP` is applied here in a spatial modeling setting (i.e., with two input variables), it was designed for computer simulation modeling and has been shown to work well in input dimension as high as ten.

3. THE COMPETITION

At the initial planning phase of this competition, we desired to compare a broad variety of approaches: from frequentist to Bayesian and from well-established to modern developments. In accordance with this plan, efforts were made to contact a variety of research groups with strong expertise in a method to analyze the datasets. After this outreach period, the research teams listed in Table 1 agreed to participate and implement their associated method.

The groups were provided with two training datasets: one real and one simulated. Both datasets consisted of observations on the same 500×300 grid ranging longitude values of -95.91153 to -91.28381 and latitude values of 34.29519 to 37.06811. The real dataset consisted of daytime land surface temperatures as measured by the Terra instrument onboard the MODIS satellite on August 4, 2016 (Level-3 data). The data was downloaded from the MODIS reprojection tool web interface

Table 1. Research groups participating in the competition along with their selected method (competitor).

Group Members	Method
Abhirup Datta & Andrew Finley	Nearest Neighbor Processes
Andrew Finley	Predictive Processes
Reinhard Furrer	Covariance Tapering
Florian Gerber	Gapfill
Raj Guhaniyogi	Metakriging
Matthew J. Heaton	Spatial Partitioning
Andrew Zammit-Mangion	Fixed rank kriging
Matthias Katzfuss & Dorit Hammerling	Multiresolution Approximations
Finn Lindgren	Stochastic Partial Differential Equations
Douglas Nychka	Lattice Kriging
Robert Gramacy & Furong Sun	Local Approximate Gaussian Processes

(MRTweb) located at `https://mrtweb.cr.usgs.gov/` and is provided as supplementary material to this article. The latitude and longitude range, as well as the date, were chosen because of the sparse cloud cover over the region on this date (rather than by scientific interest in the date itself). Namely, only 1.1% of the Level-3 MODIS data were corrupted by cloud cover leaving 148,309/150,000 observed values to use for our purposes.

The simulated dataset was created by, first, fitting a Gaussian process model with constant mean, exponential covariance function and a nugget effect to a random sample of 2500 observations from the above MODIS data. The resulting parameter estimates were then used to simulate 150,000 observations on the same grid as the MODIS data.

In order to ensure a realistic analysis scenario, the missing data pattern on August 6, 2016 from the same MODIS satellite data product was used to separate each dataset into training and test sets. After the split, the training set for the MODIS data consisted of 105,569 observations leaving 42,740 observations in the test set. The training set for the simulated data also consisted of 105,569 observations but a test set size of 44,431 (the difference in test set size is contributed to missing data due to cloud cover in the original MODIS data). Research teams were provided with the

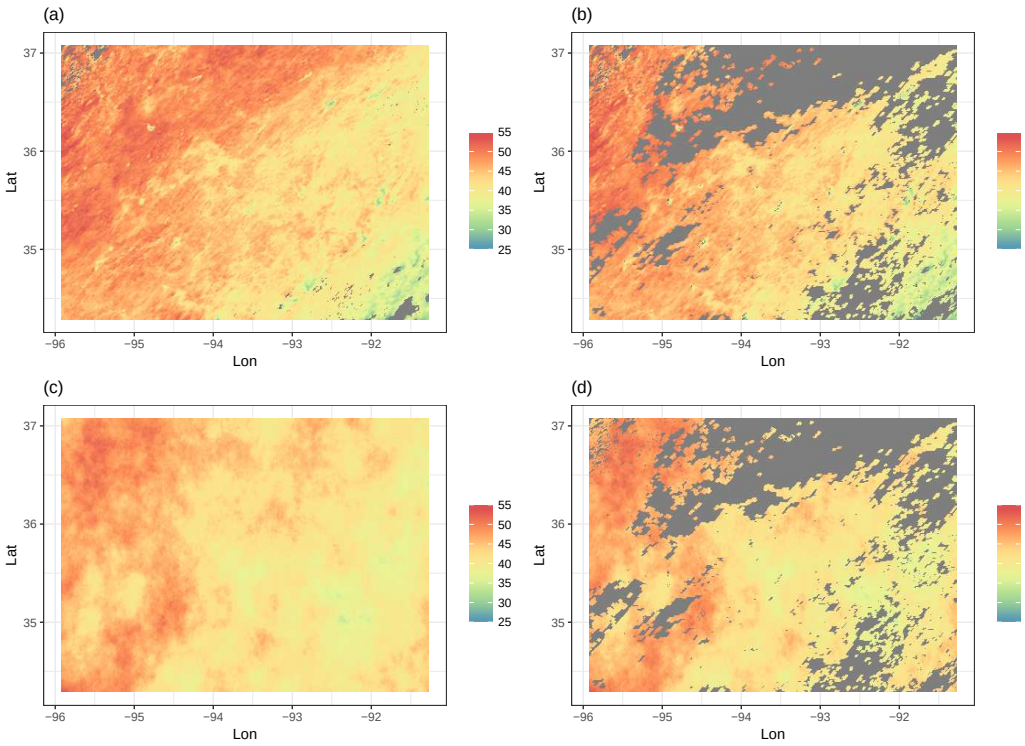

Figure 2. The top row displays the (a) full and (b) training satellite datasets. The bottom row displays the (c) full and (d) training simulated data.

training set and the locations of the test set (but not the actual observation in the test set). Figure 2 displays the full datasets along with the corresponding training set provided to each research group. All datasets used in this article are provided as supplementary material to this article.

Each group independently wrote code (all of which is included as supplementary material to this article) that provided (i) a point prediction for each location in the test set, (ii) a 95% prediction interval for location in the test set or a corresponding standard error for the prediction, (iii) the average time required to implement the method per iteration and (iv) the total clock time needed to implement the method. In order to minimize the number of confounding factors in this competition, each group was instructed to use an exponential correlation function (if applicable to their chosen method) and a nugget variance. For the simulated data the groups were instructed to only use a constant mean (because this was how the data was originally simulated). However, for the satellite data, the groups used a linear effect for latitude and longitude so that the residual process more closely resembled the exponential correlation. The code from each team was then run on the Becker computing environment (256 GB of RAM and 2 Intel Xeon E5-2680 v4 2.40GHz CPUs with 14 cores each and 2 threads per core - totaling 56 possible threads for use in parallel computing) located at Brigham Young University (BYU). Each team's code was run individually and no other processes were simultaneously run so as to provide an accurate measure of computing time.

Each method was compared in terms of mean absolute error (MAE $= n_{\text{test}}^{-1} \sum_{i=1}^{n_{\text{test}}} |y(\boldsymbol{s}_i) - \widehat{y}(\boldsymbol{s}_i)|$), root mean squared error (RMSE $= (n_{\text{test}}^{-1} \sum_{i=1}^{n_{\text{test}}} (y(\boldsymbol{s}_i) - \widehat{y}(\boldsymbol{s}_i))^2)^{1/2}$), continuous rank probability score (CRPS; see Gneiting and Raftery 2007, Gneiting and Katzfuss 2014), interval score (INT; see Gneiting and Raftery 2007) and prediction interval coverage (CVG; the percent of intervals containing the true predicted value). To calculate the CRPS, we assumed the associated predictive distribution was well approximated by a Gaussian distribution with mean centered at the predicted value and standard deviation equal to the predictive standard error. In cases where only a prediction interval was provided, the predictive standard error was taken as $(U - L)/(2 \times \Phi^{-1}(0.975))$

Table 2. Numerical scoring for each competing method on the simulated data. The best result of each score is bolded.

Method	MAE	RMSE	CRPS	INT	CVG	Run Time (Min)	Cores Used
FRK	1.03	1.31	0.74	8.35	0.84	2.18	1
Gapfill	0.73	1.00	0.64	18.01	0.44	0.63	40
Lattice Krig	0.63	0.87	0.45	4.04	0.97	25.58	1
LAGP	0.79	1.11	0.57	5.71	0.90	2.28	40
Metakriging	0.74	0.97	0.53	4.69	0.99	2888.89	30
MRA	**0.61**	**0.83**	**0.43**	**3.64**	0.93	13.57	1
NNGP Conjugate	0.65	0.88	0.46	3.79	**0.96**	1.99	1
NNGP Response	0.65	0.88	0.46	3.81	**0.96**	45.06	1
Partition	0.64	0.86	0.47	5.05	0.86	77.56	55
Pred. Proc.	0.89	1.21	0.79	12.75	0.77	639.23	1
SPDE	0.62	0.86	0.59	7.81	1.00	138.34	2
Tapering	0.69	0.97	0.55	6.39	1.00	188.36	1

where U and L are the upper and lower ends of the interval, respectively.

4. COMPETITION RESULTS

4.1 Results for Simulated Data

The numerical results for the simulated data competition are displayed in Table 2 and the associated predicted surfaces for each method are shown in Figure 3. First, consider the predictive accuracy as measured by the MAE and RMSE in Table 2. In terms of predictive accuracy, each method performed extremely well with the best MAE being 0.61 while the worst was only 1.02. Similarly, the best RMSE was 0.83 compared to a worst RMSE of only 1.33. Considering the range of the simulated data was $53.80 - 33.91 = 19.89$, a RMSE of only 1.33 is highly accurate and represents only 6.7% of the data range.

While all the methods performed well in terms of predictive accuracy, when considering uncertainty quantification some of the methods fared better than others. For example, LatticeKrig, LAGP, metakriging, MRA and NNGP all achieved near the nominal 95% coverage rate. In contrast, FRK, Gapfill, partitioning and PP achieved lower than nominal coverage while SPDE and

tapering have higher than nominal coverage. Considering UQ further, Gapfill and PP have large interval scores suggesting possible wide predictive intervals in addition to the penalty incurred from missing the true value. In this regard, it is important to keep in mind that LAGP, metakriging, MRA, NNGP and PP all can specify the "correct" exponential correlation function. Additionally, LK and SPDE have settings that can approximate the exponential correlation function well. In contrast, some methods such as FRK and Gapfill are less suited to model fields with exponential correlation functions, which may partially explain their relatively poor prediction or coverage performance in this instance.

Finally, Figure 3 displays the predictive surfaces for each method on the simulated data. The visual inspection of the predictive surfaces provides interesting insights into the various features of each method. For example, because the Gapfill method was primarily designed for spatio-temporal data, we shifted the images to create "pseudo" datasets for the Gapfill algorithm. However, this shifting resulted in a "smeared" pattern in the predictive surface which we hypothesize would not occur in the space-time setting. Likewise, arguments by Simpson et al. (2012) and Stein (2014) suggest that low rank methods oversmooth the data and such possible oversmoothing is seen in the predictive surfaces for FRK and PP.

4.2 Results for Real Data

The results for the real MODIS data are displayed in Table 3 and largely reiterate the results from the simulated data. Namely, each method performed very well in terms of predictive accuracy. The largest RMSE was only 2.52 which, when considered on the data range of $55.41 - 24.37 = 31.04$, is very small. We note that, under the setup of the competition, some of the methods were forced to approximate a GP with isotropic exponential covariance function, which is the true covariance function of the simulated data, but most certainly not for the real data. Thus, the scores are lowest for those approximations that happened to result in a good fit to the data and not necessarily lowest for those methods that best approximated the exponential covariance.

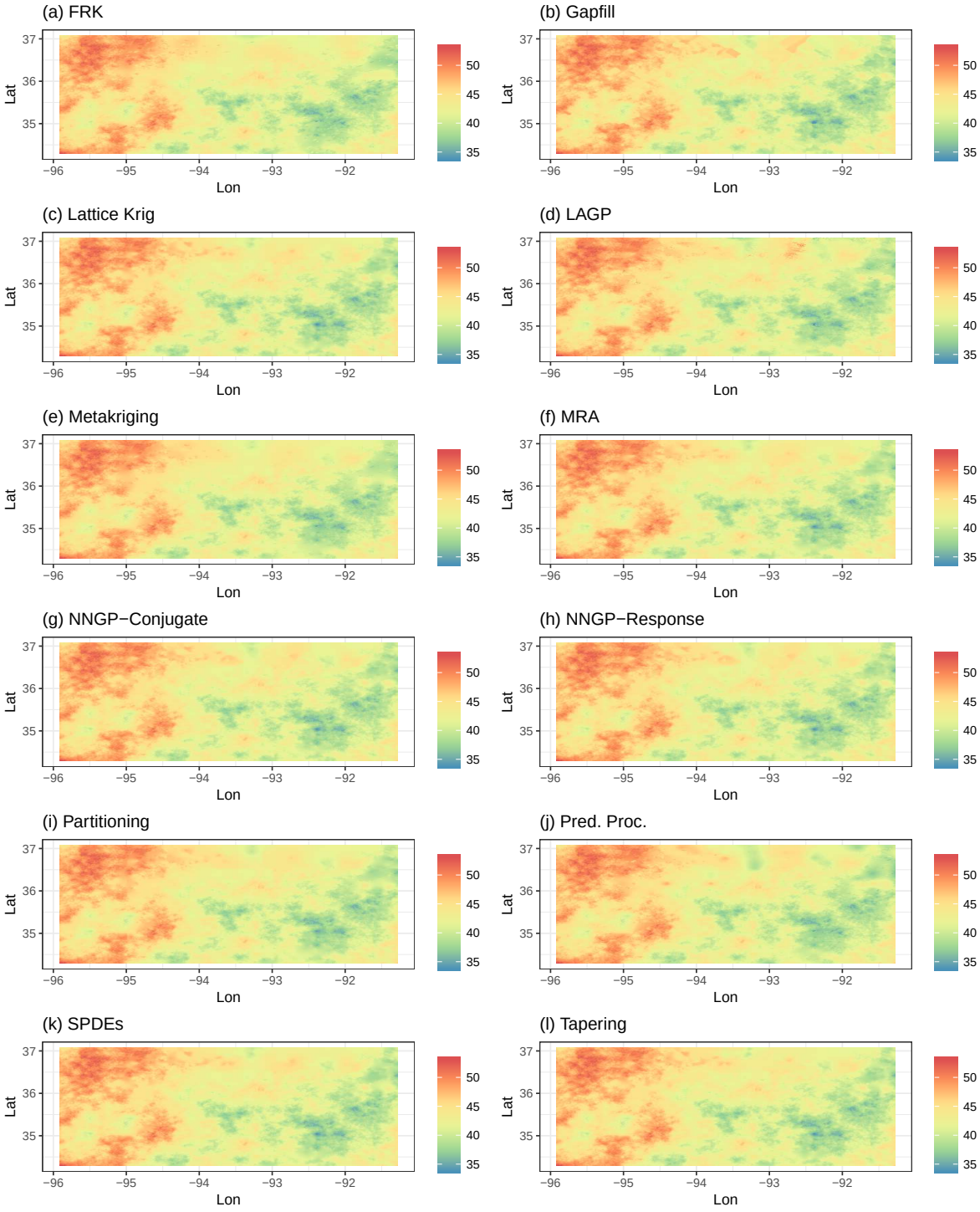

Figure 3. Predictions for the simulated data using each of the competing methods.

Table 3. Numerical scoring for each competing method on the satellite data. The best result of each score is bolded.

Method	MAE	RMSE	CRPS	INT	CVG	Run Time (Min)	Cores Used
FRK	1.96	2.44	1.44	14.08	0.79	2.32	1
Gapfill	1.33	1.86	1.17	34.78	0.36	1.39	40
Lattice Krig	1.22	1.68	0.87	7.55	0.96	27.92	1
LAGP	1.65	2.08	1.17	10.81	0.83	2.27	40
Metakriging	2.08	2.50	1.44	10.77	0.89	2888.52	30
MRA	1.33	1.85	0.94	8.00	0.92	15.61	1
NNGP Conjugate	1.21	1.64	0.85	7.57	**0.95**	2.06	1
NNGP Response	1.24	1.68	0.87	**7.50**	0.94	42.85	1
Partition	1.41	1.80	1.02	10.49	0.86	79.98	55
Pred. Proc.	2.05	2.52	1.85	26.24	0.75	640.48	1
SPDE	**1.10**	**1.53**	**0.83**	8.85	0.97	120.33	2
Tapering	1.87	2.45	1.32	10.31	0.93	133.26	1

The largest discrepancies among the competing methods is again in terms of uncertainty quantification. Lattice kriging, metakriging, MRA, NNGP again achieved near nominal coverage rates with small interval scores and CRPS. The SPDE and tapering approaches did better in terms of coverage in that the empirical rates were near nominal (recall that the corresponding coverage rates were too high for the simulated data for these methods). In contrast, the coverage rates on the MODIS data for FRK, Gapfill, LAGP, partitioning and predictive processes were too small resulting in larger interval scores.

Finally, visual inspections of the predictive surfaces for the MODIS data are shown in Figure 4. Notably the majority of the methods smooth out the predictions in the north-central region. This is to be expected because such predictions are considered "long-range" with very little (or no) observed data in this region (see Figure 2). Hence, predictions for this region rely more heavily on the overall mean surface rather than borrowing information from neighboring observations (of which there is none). Again, the "shifting" used for the Gapfill algorithm is again apparent in the predictive surface. As with the simulated data, we hypothesize that such "smeared" predictive surfaces for Gapfill would not occur under the spatio-temporal setting.

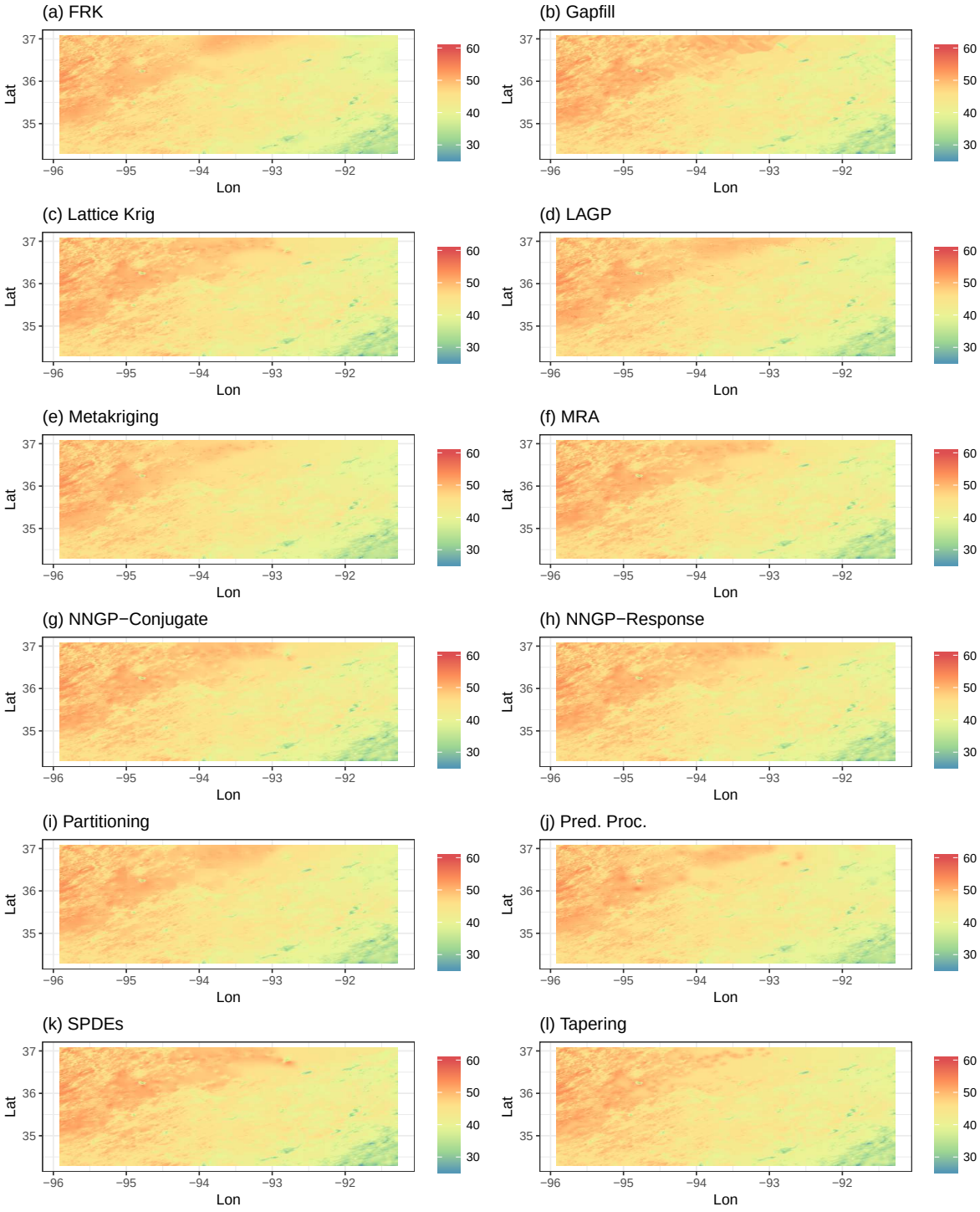

Figure 4. Predictions for the satellite data using each of the competing methods.

5. CONCLUSIONS

The contribution of this article was three-fold: (i) provide an overview of the plethora of methods available for analyzing large spatial datasets, (ii) provide a rigorous comparison of the methods based on a common task framework and (iii) make available the code to analyze the data to the broader scientific community. In terms of comparison, each of the methods performed very well in terms in predictive accuracy suggesting that any of the above methods are well suited to the task of prediction. However, the methods differed in terms of their ability to accurately quantify the uncertainty associated with the predictions. While we saw that some methods did consistently well in both predictive performance and nominal coverage on the simulated and real data, in general we can expect performance of any method to change with size of the dataset, measurement error variance, and the nature of missingness. However, the data scenario's considered here are relatively representative of a typical spatial analysis such that our results can be used as a guide for practitioners.

At the outset of this study, run time and computation time for each method was of interest. However, because many of these methods are very young in their use and implementation, the variability across run time was too great to be used as a measure to compare the methods. For example, some methods are implemented in R while others are implemented in MATLAB. Still, others use R as a front end to call C-optimized functions. Hence, while we reported the run times in the results section, we provide these as more of an "off the shelf" run time estimate rather than an optimized run time. Until time allows for each method to be further developed and software becomes available comparing run times can be misleading.

Importantly, no effort was made to standardize the time spent on this project by each group. Some groups were able to quickly code up their analysis from existing R or MATLAB libraries. Others, however, had to spend more time writing code specific to this analysis. Undoubtedly, some groups likely spent more time running "in house" cross-validation studies to validate their

model predictions prior to the final run on the BYU servers while others did not. Because of this difference, we note that some of the discrepancies in results seen here may be attributable to the amount of effort expended by each group. However, we still feel that the results displayed herein give valuable insight into the strengths and weaknesses of each method.

This comparison focused solely on spatial data. Hence, we stress that the results found here are applicable only to the spatial setting. However, spatio-temporal data are often considerably larger and more complex than spatial data. Many of the above methods have extensions to the space time setting (e.g., Gapfill is built directly for spatio-temporal settings). Further research is needed to compare these methods in the spatio-temporal setting.

■REFERENCES REFERENCES

Anderson, C., Lee, D., and Dean, N. (2014), "Identifying clusters in Bayesian disease mapping," *Biostatistics*, 15, 457–469.

Banerjee, S., Carlin, B. P., and Gelfand, A. E. (2014), *Hierarchical modeling and analysis for spatial data*, Crc Press.

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

Barbian, M. H. and Assunção, R. M. (2017), "Spatial subsemble estimator for large geostatistical data," *Spatial Statistics*, 22, 68–88.

Bevilacqua, M., Faouzi, T., Furrer, R., and Porcu, E. (2016), "Estimation and Prediction using Generalized Wendland Covariance Function under Fixed Domain Asymptotics," ArXiv:1607.06921v2.

Bradley, J. R., Cressie, N., Shi, T., et al. (2016), "A comparison of spatial predictors when datasets could be very large," *Statistics Surveys*, 10, 100–131.

Cohn, D. A. (1996), "Neural Network Exploration Using Optimal Experimental Design," in *Advances in Neural Information Processing Systems*, Morgan Kaufmann Publishers, vol. 6(9), pp. 679–686.

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

Cressie, N. and Johannesson, G. (2006), "Spatial prediction for massive data sets," in *Mastering the Data Explosion in the Earth and Environmental Sciences: Proceedings of the Australian Academy of Science Elizabeth and Frederick White Conference*, Canberra, Australia: Australian Academy of Science, pp. 1–11.

— (2008), "Fixed rank kriging for very large spatial data sets," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70, 209–226.

Cressie, N. and Wikle, C. K. (2015), *Statistics for spatio-temporal data*, John Wiley & Sons.

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

— (2016b), "On nearest-neighbor Gaussian process models for massive spatial data," *Wiley Interdisciplinary Reviews: Computational Statistics*, 8, 162–171.

Datta, A., Banerjee, S., Finley, A. O., Hamm, N. A., Schaap, M., et al. (2016c), "Nonseparable dynamic nearest neighbor Gaussian process models for large spatio-temporal data with an application to particulate matter analysis," *The Annals of Applied Statistics*, 10, 1286–1316.

Du, J., Zhang, H., and Mandrekar, V. S. (2009), "Fixed-domain asymptotic properties of tapered maximum likelihood estimators," *Ann. Statist.*, 37, 3330–3361.

Eidsvik, J., Shaby, B. A., Reich, B. J., Wheeler, M., and Niemi, J. (2014), "Estimation and prediction in spatial models with block composite likelihoods," *Journal of Computational and Graphical Statistics*, 23, 295–315.

Emery, X. (2009), "The kriging update equations and their application to the selection of neighboring data," *Computational Geosciences*, 13, 269–280.

Finley, A., Datta, A., and Banerjee, S. (2017a), *spNNGP: Spatial Regression Models for Large Datasets using Nearest Neighbor Gaussian Processes*, r package version 0.1.1.

Finley, A. O., Datta, A., Cook, B. C., Morton, D. C., Andersen, H. E., and Banerjee, S. (2017b), "Applying Nearest Neighbor Gaussian Processes to Massive Spatial Data Sets: Forest Canopy Height Prediction Across Tanana Valley Alaska," .

Finley, A. O., Sang, H., Banerjee, S., and Gelfand, A. E. (2009), "Improving the performance of predictive process modeling for large datasets," *Computational statistics & data analysis*, 53, 2873–2884.

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

Furrer, R. (2016), *spam: SPArse Matrix*, r package version 1.4-0.

Furrer, R., Bachoc, F., and Du, J. (2016), "Asymptotic Properties of Multivariate Tapering for Estimation and Prediction," *J. Multivariate Anal.*, 149, 177–191.

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.

Furrer, R. and Sain, S. R. (2010), "spam: A Sparse Matrix R Package with Emphasis on MCMC Methods for Gaussian Markov Random Fields," *J. Stat. Softw.*, 36, 1–25.

Gerber, F. (2017), *gapfill: Fill Missing Values in Satellite Data*, r package version 0.9.5.

Gerber, F., Furrer, R., Schaepman-Strub, G., de Jong, R., and Schaepman, M. E. (2016), "Predicting missing values in spatio-temporal satellite data," *ArXiv e-prints*.

Gneiting, T. and Katzfuss, M. (2014), "Probabilistic forecasting," *Annual Review of Statistics and Its Application*, 1, 125–151.

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

Gramacy, R. and Apley, D. (2015), "Local Gaussian Process Approximation for Large Computer Experiments," *Journal of Computational and Graphical Statistics*, 24, 561–578.

Gramacy, R., Niemi, J., and Weiss, R. (2014), "Massively Parallel Approximate Gaussian Process Regression," *Journal of Uncertainty Quantification*, 2, 564–584.

Gramacy, R. B. (2016), "laGP: Large-Scale Spatial Modeling via Local Approximate Gaussian Processes in R," *Journal of Statistical Software*, 72, 1–46.

Gramacy, R. B. and Haaland, B. (2016), "Speeding up neighborhood search in local Gaussian process prediction," *Technometrics*, 58, 294–303.

Guhaniyogi, R. and Banerjee, S. (2017), "Meta-Kriging: scalable Bayesian modeling and inference for massive spatial datasets," *https://www.soe.ucsc.edu/research/technical-reports/UCSC-SOE-17-07*.

Heaton, M. J., Christensen, W. F., and Terres, M. A. (2017), "Nonstationary Gaussian process models using spatial hierarchical clustering from finite differences," *Technometrics*, 59, 93–101.

Higdon, D. (2002), "Space and space-time modeling using process convolutions," in *Quantitative methods for current environmental issues*, Springer, pp. 37–56.

Hirano, T. and Yajima, Y. (2013), "Covariance tapering for prediction of large spatial data sets in transformed random fields," *Annals of the Institute of Statistical Mathematics*, 65, 913–939.

Kang, E. L. and Cressie, N. (2011), "Bayesian inference for the spatial random effects model," *Journal of the American Statistical Association*, 106, 972–983.

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

Katzfuss, M. and Cressie, N. (2011), "Spatio-temporal smoothing and EM estimation for massive remote-sensing data sets," *Journal of Time Series Analysis*, 32, 430–446.

Katzfuss, M. and Hammerling, D. (2017), "Parallel inference for massive distributed spatial data using low-rank models," *Statistics and Computing*, 27, 363–375.

Kaufman, C. G., Schervish, M. J., and Nychka, D. W. (2008), "Covariance tapering for likelihood-based estimation in large spatial data sets," *Journal of the American Statistical Association*, 103, 1545–1555.

Kim, H.-M., Mallick, B. K., and Holmes, C. (2005), "Analyzing nonstationary spatial data using piecewise Gaussian processes," *Journal of the American Statistical Association*, 100, 653–668.

Knorr-Held, L. and Raßer, G. (2000), "Bayesian detection of clusters and discontinuities in disease maps," *Biometrics*, 56, 13–21.

Konomi, B. A., Sang, H., and Mallick, B. K. (2014), "Adaptive bayesian nonstationary modeling for large spatial datasets using covariance approximations," *Journal of Computational and Graphical Statistics*, 23, 802–829.

Lemos, R. T. and Sansó, B. (2009), "A spatio-temporal model for mean, anomaly, and trend fields of North Atlantic sea surface temperature," *Journal of the American Statistical Association*, 104, 5–18.

Liang, F., Cheng, Y., Song, Q., Park, J., and Yang, P. (2013), "A resampling-based stochastic approximation method for analysis of large geostatistical data," *Journal of the American Statistical Association*, 108, 325–339.

Lindgren, F., Rue, H., and Lindström, J. (2011), "An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73, 423–498.

Minsker, S. (2015), "Geometric median and robust estimation in Banach spaces," *Bernoulli*, 21, 2308–2335.

Minsker, S., Srivastava, S., Lin, L., and Dunson, D. B. (2014), "Robust and scalable Bayes via a median of subset posterior measures," *arXiv preprint arXiv:1403.2660*.

Neelon, B., Gelfand, A. E., and Miranda, M. L. (2014), "A multivariate spatial mixture model for areal data: examining regional differences in standardized test scores," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 63, 737–761.

Nychka, D., Bandyopadhyay, S., Hammerling, D., Lindgren, F., and Sain, S. (2015), "A multiresolution Gaussian process model for the analysis of large spatial datasets," *Journal of Computational and Graphical Statistics*, 24, 579–599.

Paciorek, C. J., Lipshitz, B., Zhuo, W., Kaufman, C. G., Thomas, R. C., et al. (2013), "Parallelizing Gaussian Process Calculations In R," *arXiv preprint arXiv:1305.4886*.

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

Rue, H., Martino, S., Lindgren, F., Simpson, D., Riebler, A., Krainski, E. T., and Fuglstad, G.-A.

(2017), *INLA: Bayesian Analysis of Latent Gaussian Models using Integrated Nested Laplace Approximations*, r package version 17.06.20.

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

Schabenberger, O. and Gotway, C. A. (2004), *Statistical methods for spatial data analysis*, CRC press.

Simpson, D., Lindgren, F., and Rue, H. (2012), "In order to make spatial statistics computationally feasible, we need to forget about the covariance function," *Environmetrics*, 23, 65–74.

Stein, M. L. (1999), *Interpolation of Spatial Data*, Springer-Verlag, some theory for Kriging.

— (2013), "Statistical properties of covariance tapers," *Journal of Computational and Graphical Statistics*, 22, 866–885.

— (2014), "Limitations on low rank approximations for covariance matrices of spatial data," *Spatial Statistics*, 8, 1–19.

Stein, M. L., Chi, Z., and Welty, L. J. (2004), "Approximating likelihoods for large spatial data sets," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 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*, Springer, pp. 55–77.

Vapnik, V. (1995), *The Nature of Statistical Learning Theory*, New York: Springer Verlag.

Varin, C., Reid, N., and Firth, D. (2011), "An overview of composite likelihood methods," *Statistica Sinica*, 5–42.

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

Wang, D. and Loh, W.-L. (2011), "On fixed-domain asymptotics and covariance tapering in Gaussian random field models," *Electron. J. Statist.*, 5, 238–269.

Weiss, D. J., Atkinson, P. M., Bhatt, S., Mappin, B., Hay, S. I., and Gething, P. W. (2014), "An effective approach for gap-filling continental scale remotely sensed time-series," *ISPRS J. Photogramm. Remote Sens.*, 98, 106–118.

Wikle, C. K., Cressie, N., Zammit-Mangion, A., and Shumack, C. (2017), "A Common Task Framework (CTF) for Objective Comparison of Spatial Prediction Methodologies," *Statistics Views*.

Zammit-Mangion, A. and Cressie, N. (2017), "FRK: An R Package for Spatial and Spatio-Temporal Prediction with Large Datasets," *arXiv preprint arXiv:1705.08105*.