Journal of Statistical Software

April 2007, Volume 19, Issue 4. *http://www.jstatsoft.org/*

spBayes: An R Package for Univariate and Multivariate Hierarchical Point-referenced Spatial Models

Andrew O. Finley
University of Minnesota

Sudipto Banerjee
University of Minnesota

Bradley P. Carlin
University of Minnesota

Abstract

Scientists and investigators in such diverse fields as geological and environmental sciences, ecology, forestry, disease mapping, and economics often encounter spatially referenced data collected over a fixed set of locations with coordinates (latitude–longitude, Easting–Northing etc.) in a region of study. Such point-referenced or geostatistical data are often best analyzed with Bayesian hierarchical models. Unfortunately, fitting such models involves computationally intensive Markov chain Monte Carlo (MCMC) methods whose efficiency depends upon the specific problem at hand. This requires extensive coding on the part of the user and the situation is not helped by the lack of available software for such algorithms. Here, we introduce a statistical software package, **spBayes**, built upon the R statistical computing platform that implements a generalized template encompassing a wide variety of Gaussian spatial process models for univariate as well as multivariate point-referenced data. We discuss the algorithms behind our package and illustrate its use with a synthetic and real data example.

Keywords: Bayesian inference, coregionalization, kriging, Markov chain Monte Carlo, multivariate spatial process, R.

1. Introduction

Recent advances in Geographical Information Systems (GIS) and Global Positioning Systems (GPS) have led to increased interest in modeling and analysis of geocoded data arising in scientific research. This interest has, in turn, led to significant developments in such modeling; see, for example, the books by Cressie (1993), Chilés and Delfiner (1999), Møller (2003), Schabenberger and Gotway (2004), and Banerjee *et al.* (2004) for a variety of methods and applications. Two underlying configurations are encountered commonly in practice: locations that are areas or regions with well–defined neighbors (such as pixels in a lattice, counties in a map, etc.), called *areally referenced* data, and locations that are points with coordinates (latitude–longitude, Easting–Northing, etc.), termed *point-referenced* or *geostatistical*. Statistical modeling approaches differ depending upon the underlying configuration: for areal data, one seeks to build models using conditional independence assumptions based upon the neighborhood or adjacency structure of the regions, while for the geostatistical setting, one incorporates spatial correlations as decaying continuously with direction and distance.

It is also well recognized in the statistics literature that spatial associations are captured most effectively using *hierarchical* models that build dependencies in different stages. These models follow the Bayesian paradigm of statistical inference (see e.g., Carlin and Louis 2000; Gelman *et al.* 2004), where analysis is based upon sampling from the posterior distributions of the different model parameters. Hierarchical models are especially advantageous with data sets having several lurking sources of variation and dependence, where they can estimate much richer models with less stringent assumptions.

Recent computational advances with regard to Markov chain Monte Carlo (MCMC) methods have contributed enormously to the popularity of hierarchical models in a wide array of disciplines (e.g., Gilks *et al.* 1996), and spatial modeling is no exception (see e.g., Banerjee *et al.* 2004). In the realm of spatial statistics, hierarchical models have been widely applied to analyze both areally referenced as well as point-referenced or geostatistical data. For the former, a class of models known as Conditionally Autoregressive (CAR) models have become very popular as they are easily implemented using MCMC methods such as the Gibbs sampler. In fact, these models are somewhat naturally suited for the Gibbs sampler which draws samples from conditional distributions that are fully specified by the CAR models. Their popularity has increased in no small measure also due to their automated implementation in the **WinBUGS** software package. This is an offshoot of the **BUGS** (Bayesian inference Using Gibbs Sampling) project for the Windows platform and provides a flexible and user-friendly interface to construct hierarchical models that are implemented using a Gibbs sampler. This is performed by identifying an hierarchical model with a *directed acyclic graph* (DAG) whose nodes form the different components of the model and allow the language identify the full conditional distributions that need to be updated.

From an automated implementation perspective, however, the state of affairs is less encouraging for point-referenced models. While **BUGS** can be used to build such models, its scope is somewhat limited. First, these models involve relatively expensive matrix computations that can become prohibitive with large data sets. Second, the routines fit unmarginalized models which are less suited for direct updating using a Gibbs sampler in the **BUGS** paradigm and results in slower convergence of the chains. Thirdly, investigators often encounter multivariate spatial data sets with several spatially dependent responses, whose analysis requires multivariate spatial models that involve matrix computations that are poorly implemented in **BUGS**. Several other models for point-referenced spatial data analysis (such as non–stationary models, spatially varying regression models, multi–resolution models, and spatiotemporal models) are difficult to implement in **BUGS** and require specialized programming. In particular, lower–level languages such as C/C++ and FORTRAN are needed in conjunction with efficient matrix computation libraries such as **BLAS** (Basic Linear Algebra Subprograms, see Blackford *et al.* 2002) and **LAPACK** (Linear Algebra Package, see Anderson *et al.* 1999) to implement these models. **BLAS** and **LAPACK** subroutines and documentation are available at `http://www.netlib.org/blas/` and `http://www.netlib.org/lapack/`, respectively.

An exciting development in bringing bring such sophisticated statistical methodology to users is the R project (R Development Core Team 2007). Not only does R offer an environment with several built-in functions for mathematical computations, it also provides an edifice for developers to build packages (or libraries) offering specialized functions. These packages can be written in lower–level languages for optimizing performance, while maintaining a friendly user-interface.

Several R packages that facilitate spatial modeling exist, but most of them do not implement Bayesian hierarchical models. The most notable exceptions are **geoR** (Ribeiro, Jr. and Diggle 2001) and **geoRglm** (Christensen and Ribeiro, Jr. 2002) which implement Bayesian spatial Gaussian and generalized linear regression models, respectively. In addition to handling only the simplest spatial regression models with a single dependent variable, these packages do not provide a full Bayesian approach, opting rather to discretize some prior distributions for computational purposes.

This manuscript discusses a generalized template that can be used to fit a wide variety of spatial process models for univariate as well as multivariate point-referenced data. We discuss the design of our R package **spBayes** that implements this template using efficient MCMC algorithms. For the current article, we restrict ourselves to the Gaussian response setting as our focus is on the wide variety of Gaussian process models for spatial data analysis. Section 2 discusses spatial regression models arising in multivariate process contexts. Next, in Section 3 we outline the generalized template we propose to implement these models and explain how we carry out inference and spatial predictions in a sampling–based framework. The R package we envision for implementing this template is described in Section 4 along with two illustrations. Finally, Section 5 concludes the paper and indicates upcoming work.

2. Spatial regression models

Univariate spatial regression models for point-referenced data assume a response or dependent variable $Y(\mathbf{s})$ observed at a generic location $\mathbf{s}$ along with a $p \times 1$ vector of spatially referenced predictors $\mathbf{x}(\mathbf{s})$. The regression model is written as

$$Y(\mathbf{s}) = \mathbf{x}^\top(\mathbf{s})\boldsymbol{\beta} + W(\mathbf{s}) + \epsilon(\mathbf{s}), \tag{1}$$

where $\{W(s) : s \in \mathscr{D}\}$ is a *spatial random field*, with $\mathscr{D}$ an open subset of $\Re^d$ of dimension d; in most practical settings $d = 2$ or $d = 3$. We say that a random field is a *valid* spatial process if for an any finite collection of sites $\mathscr{S} = \{\mathbf{s}_1, \ldots, \mathbf{s}_n\}$ of arbitrary size, the vector $\mathbf{W} = [W(\mathbf{s}_i)]_{i=1}^n$ follows a well-defined joint probability distribution. Also, $\epsilon(\mathbf{s}) \overset{iid}{\sim} N(0, \tau^2)$ is a *white-noise* process, often called the *nugget* effect, modelling measurement error or micro-scale variation.

We extend the univariate case to the multivariate spatial regression model when each site $\mathbf{s}$ offers an $m \times 1$ response vector $\mathbf{Y}(\mathbf{s}) = [Y_i(\mathbf{s})]_{i=1}^m$ along with an $m \times q$ matrix of regressors $\mathbf{X}^\top(\mathbf{s}) = [\mathbf{x}_i^\top(\mathbf{s})]_{i=1}^m$, where $q = \sum_{i=1}^m p_i$. Further, in the multivariate setting, $\mathbf{W}(\mathbf{s}) = [W_i(\mathbf{s})]_{i=1}^m$ is an $m \times 1$ zero-centered *multivariate Gaussian Process*, denoted $\mathbf{W}(\mathbf{s}) \sim MVGP(\mathbf{0}, \mathbf{K}(\cdot, \cdot; \boldsymbol{\theta}))$ and capturing spatial variation, and $\boldsymbol{\epsilon}(\mathbf{s}) \sim MVN(\mathbf{0}, \Psi)$ models the measurement error effect for the response with the $m \times m$ dispersion matrix Ψ. The multivariate Gaussian process is completely specified by an $m \times m$ *cross–covariance* matrix function $\mathbf{K}(\mathbf{s}, \mathbf{s}'; \boldsymbol{\theta}) = [Cov(W_i(\mathbf{s}), W_j(\mathbf{s}'))]_{i,j=1}^m$ whose (i, j)-th element is the covariance between $W_i(\mathbf{s})$

and $W_j(\mathbf{s}')$, with $\boldsymbol{\theta}$ being certain parameters that may control correlation decay and smoothness of the process. Then, for any integer n and any collection of sites $\mathscr{S}$, the $mn \times 1$ vector $\mathbf{W} = [\mathbf{W}(\mathbf{s}_i)]_{i=1}^n$ is distributed as a multivariate normal distribution $\mathbf{W} \sim MVN(\mathbf{0}, \Sigma_{\mathbf{W}}(\boldsymbol{\theta}))$, known as a *realization* of the spatial process. Here $\Sigma_{\mathbf{W}}(\boldsymbol{\theta}) = [\mathbf{K}(\mathbf{s}_i, \mathbf{s}_j; \boldsymbol{\theta})]_{i,j=1}^n$ is the $mn \times mn$ matrix with $\mathbf{K}(\mathbf{s}_i, \mathbf{s}_j; \boldsymbol{\theta})$ forming the (i,j)-th $m \times m$ block. The covariance matrix of the observed response vector $\mathbf{Y} = [\mathbf{Y}(\mathbf{s}_i)]_{i=1}^n$ is $\Sigma_{\mathbf{W}}(\boldsymbol{\theta}) + I_n \otimes \Psi$, where I_n is the $n \times n$ identity matrix and $\otimes$ denotes the Kronecker product (e.g., Harville 1997).

Care is needed in choosing $\mathbf{K}(\mathbf{s}, \mathbf{s}'; \boldsymbol{\theta})$ so that $\Sigma_{\mathbf{W}}(\boldsymbol{\theta})$ is symmetric and positive definite. Modeling $K(\mathbf{s}, \mathbf{s}'; \boldsymbol{\theta})$ is indeed more demanding than choosing real-valued covariance functions in univariate spatial modeling that are characterized by Bochner's Theorem (see e.g., Cressie 1993, p. 84). In the multivariate setting, we require that for an arbitrary number and choice of locations, the resulting $\Sigma_{\mathbf{W}}(\boldsymbol{\theta})$ be symmetric and positive definite. Note that the cross-covariance matrix function need not be symmetric or positive definite but must satisfy $\mathbf{K}(\mathbf{s}', \mathbf{s}; \boldsymbol{\theta}) = \mathbf{K}^\top(\mathbf{s}, \mathbf{s}'; \boldsymbol{\theta})$ so that $\Sigma_{\mathbf{W}}(\boldsymbol{\theta})$ is symmetric. In the limiting sense, as $\mathbf{s}' \to \mathbf{s}$, $\mathbf{K}(\mathbf{s}, \mathbf{s}; \boldsymbol{\theta}) = [Cov(W_i(\mathbf{s}), W_j(\mathbf{s}))]_{i,j=1}^m$ becomes the symmetric and positive definite variance-covariance matrix of $\mathbf{W}(\mathbf{s})$ *within* site $\mathbf{s}$. A theorem by Cramér (see e.g., Chilés and Delfiner 1999) characterizes cross-covariance functions, akin to Bochner's theorem for univariate covariance functions, but using Cramér's result in practical modeling is trivial. Recent work by Majumdar and Gelfand (2006) reviews other cross-covariance modeling approaches, such as averaging and convolving correlation functions, but point out the computational and modeling difficulties involved.

Since our primary objective is to develop a computationally feasible template that accommodates sufficiently rich multivariate spatial models, we adopt a constructive approach through *coregionalization* models (Wackernagel 2003). To motivate this approach, one considers simpler cross-covariance functions and builds richer models by linearly transforming them. For instance, let $\tilde{\mathbf{W}}(\mathbf{s}) = [\tilde{W}_i(\mathbf{s})]_{i=1}^m$ be an $m \times 1$ process with independent zero-centered spatial processes with unit variance; that is, each $\tilde{W}_i(\mathbf{s}) \sim GP(0, \rho(\cdot, \cdot))$ with $Var(\tilde{W}_k(\mathbf{s})) = 1$ and $Cov(\tilde{W}_i(\mathbf{s}), \tilde{W}_i(\mathbf{s}')) = \rho_i(\mathbf{s}, \mathbf{s}'; \boldsymbol{\theta}_i)$ and $Cov(\tilde{W}_i(\mathbf{s}), \tilde{W}_j(\mathbf{s}')) = 0$ whenever $i \neq j$ (irrespective of how close $\mathbf{s}$ and $\mathbf{s}'$ are), where $\rho_i(\cdot; \boldsymbol{\theta}_i)$ is a correlation function associated with $\tilde{W}_i(\mathbf{s})$ and $\boldsymbol{\theta}_i$ are parameters therein. This yields a diagonal cross-covariance matrix function $\tilde{\mathbf{K}}(\mathbf{s}, \mathbf{s}'; \boldsymbol{\theta}) = diag[\rho_i(\mathbf{s}, \mathbf{s}'; \boldsymbol{\theta}_i)]_{i=1}^m$ with $\boldsymbol{\theta} = \{\boldsymbol{\theta}_i\}_{i=1}^m$. It is easy to verify that $\tilde{\mathbf{K}}(\mathbf{s}, \mathbf{s}'; \boldsymbol{\theta})$ is a valid cross-covariance matrix.

The Matérn correlation function allows control of spatial association and smoothness (see e.g., Stein 1999) and is given by

$$\rho(\mathbf{s}, \mathbf{s}'; \phi, \nu) = \frac{1}{2^{\nu-1}\Gamma(\nu)}(\|\mathbf{s} - \mathbf{s}'\|\phi)^\nu \mathscr{K}_\nu(\|\mathbf{s} - \mathbf{s}'\|; \phi); \quad \phi > 0, \ \nu > 0, \tag{2}$$

where ϕ controls the decay in spatial correlation and ν is a smoothness parameter with higher values yielding smoother process realizations. Also, Γ is the usual Gamma function while $\mathscr{K}_\nu$ is a modified Bessel function of the third kind with order ν, and $\|\mathbf{s} - \mathbf{s}'\|$ is the Euclidean distance between sites $\mathbf{s}$ and $\mathbf{s}'$. Covariance functions that depend upon the distance metric only are often referred to as *isotropic*. Several other choices for valid correlation functions are discussed in Banerjee *et al.* (2004). For $\tilde{W}_i(\mathbf{s})$ we choose isotropic Matérn functions $\rho_i(\mathbf{s}, \mathbf{s}'; \boldsymbol{\theta}_i)$ with $\boldsymbol{\theta}_i = (\phi_i, \nu_i)$ for $i = 1, \ldots, m$.

For building richer covariance structures, we assume the process $\mathbf{W}(\mathbf{s}) = \mathbf{A}(\mathbf{s})\tilde{\mathbf{W}}(\mathbf{s})$ to be a linear transformation of $\tilde{\mathbf{W}}(\mathbf{s})$, where $\mathbf{A}(\mathbf{s})$ is a space-varying matrix transform that is nonsingular for all $\mathbf{s}$. Then, the cross-covariance matrix functions are related as $\mathbf{K}(\mathbf{s},\mathbf{s}';\boldsymbol{\theta}) = \mathbf{A}(\mathbf{s})\tilde{\mathbf{K}}(\mathbf{s},\mathbf{s}')\mathbf{A}^{\top}(\mathbf{s}')$. It is worth noting that $\tilde{\mathbf{K}}(\mathbf{s},\mathbf{s};\boldsymbol{\theta}) = I_m$ (the $m \times m$ identity matrix), so that $\mathbf{K}(\mathbf{s},\mathbf{s};\boldsymbol{\theta}) = \mathbf{A}(\mathbf{s})\mathbf{A}^{\top}(\mathbf{s})$. Therefore $\mathbf{A}(\mathbf{s}) = \mathbf{K}^{1/2}(\mathbf{s},\mathbf{s};\boldsymbol{\theta})$ is identified as a Cholesky square-root of $\mathbf{K}(\mathbf{s},\mathbf{s};\boldsymbol{\theta})$ and can be taken to be lower–triangular without loss of generality. Indeed, the one-one correspondence between the elements of the Cholesky square-root matrix and the original matrix is well known (see e.g., Harville 1997, p. 229). Turning to the validity of $\mathbf{K}(\mathbf{s},\mathbf{s}';\boldsymbol{\theta})$, we argue that since $\tilde{\mathbf{K}}(\mathbf{s},\mathbf{s}';\boldsymbol{\theta})$ is a valid cross-covariance matrix, so is $\mathbf{K}(\mathbf{s},\mathbf{s}';\boldsymbol{\theta})$. To see this, first observe that the dispersion matrix of realizations of $\mathbf{W}(\mathbf{s})$ over $\mathscr{S}$, $\Sigma_{\mathbf{W}} = [\mathbf{K}(\mathbf{s}_i,\mathbf{s}_j;\boldsymbol{\theta})]_{i,j=1}^{n}$, can be written as:

$$\begin{aligned}
[\mathbf{A}(\mathbf{s}_i)\tilde{\mathbf{K}}(\mathbf{s}_i,\mathbf{s}_j;\boldsymbol{\theta})\mathbf{A}^{\top}(\mathbf{s}_j)]_{i,j=1}^{n} &= [\oplus_{i=1}^{k}\mathbf{A}(\mathbf{s}_i)][\oplus_{k=1}^{m}\rho_k(\mathbf{s}_i,\mathbf{s}_j;\boldsymbol{\theta}_k)]_{i,j=1}^{n}[\oplus_{i=1}^{k}\mathbf{A}^{\top}(\mathbf{s}_i)] \\
&= \mathscr{A}\Sigma_{\tilde{\mathbf{W}}}\mathscr{A}^{\top}, \qquad (3)
\end{aligned}$$

where $\oplus$ is the "diagonal" or direct-sum matrix operator (e.g., Harville 1997). Thus, $\oplus_{k=1}^{m}\rho_k(\mathbf{s}_i,\mathbf{s}_j;\boldsymbol{\theta}_k)$ is an $m \times m$ diagonal matrix with $\rho_k(\mathbf{s}_i,\mathbf{s}_j;\boldsymbol{\theta})$ as its diagonals while $\mathscr{A}$ is a block-diagonal matrix with the i-th diagonal block being $\mathbf{A}(\mathbf{s}_i)$. Since $\tilde{\mathbf{K}}(\mathbf{s}_i,\mathbf{s}_j;\boldsymbol{\theta})$ is a valid cross-covariance, $\Sigma_{\tilde{\mathbf{W}}}$ is positive–definite and so is $\Sigma_{\mathbf{W}}$ (by virtue of 3).

Stationary cross-covariance functions necessarily imply the linear transformation to be independent of space. Here, since the cross-covariance is a function of the separation between sites, we have $\mathbf{K}(\mathbf{s},\mathbf{s};\boldsymbol{\theta}) = \mathbf{K}(\mathbf{0};\boldsymbol{\theta})$ so that $\mathbf{A}(\mathbf{s}) = \mathbf{A} = \mathbf{K}^{1/2}(\mathbf{0};\boldsymbol{\theta})$. In such cases, $\mathscr{A} = I \otimes \mathbf{A}$ and (3) reduces to

$$\Sigma_{\mathbf{W}} = (I_n \otimes \mathbf{A})\Sigma_{\tilde{\mathbf{W}}}(I_n \otimes \mathbf{A}^{\top}). \qquad (4)$$

As a further simplification, suppose we choose $\tilde{\mathbf{K}}(\mathbf{s},\mathbf{s}';\boldsymbol{\theta}) = \rho(\mathbf{s}-\mathbf{s}';\boldsymbol{\theta})I_m$, i.e., a single correlation function for each component of $\tilde{\mathbf{W}}(\mathbf{s})$. This yields $\Sigma_{\tilde{\mathbf{W}}} = R(\boldsymbol{\theta}) \otimes I_m$, where $R(\boldsymbol{\theta}) = [\rho(\mathbf{s}_i,\mathbf{s}_j;\boldsymbol{\theta})]_{i,j=1}^{n}$ and results in a *separable* or *intrinsic* specification (see e.g., Wackernagel 2003):

$$\Sigma_{\mathbf{W}} = (I_n \otimes \mathbf{A})(R \otimes I_m)(I_n \otimes \mathbf{A}^{\top}) = R(\boldsymbol{\theta}) \otimes \mathbf{K}(\mathbf{0};\boldsymbol{\theta}). \qquad (5)$$

Here, the dispersion structure separates into a spatial component $R(\boldsymbol{\theta})$ and a within-site dispersion matrix $\mathbf{K}(\mathbf{0};\boldsymbol{\theta})$. While such models have nicer interpretability, they are often too simplistic, resulting in poorer fits.

3. Bayesian implementation using a generalized template

3.1. Estimation of model parameters

We adopt a Bayesian approach specifying prior distributions on the parameters to build hierarchical models that are estimated using a Gibbs sampler, with Metropolis–Hastings updates when required, for fitting our models (see e.g., Gelman *et al.* 2004, Chapter 11). Although such algorithms are usually problem–specific, often requiring intensive coding, casting the problem in a general template allows several models to be fit without rewriting vast amounts of code. We cast the data model into the following generic template:

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \mathscr{A}\tilde{\mathbf{W}} + \boldsymbol{\epsilon};\ \boldsymbol{\epsilon} \sim N(\mathbf{0}, I_m \otimes \Psi), \qquad (6)$$

where $\mathbf{Y}$ is the $mn \times 1$ response vector, $\mathbf{X}$ is the $mn \times q$ matrix of regressors, and $\boldsymbol{\beta}$ is the corresponding vector of regression coefficients. The specifications for the $mn \times mn$ matrices

$\mathscr{A}$ and $\tilde{\mathbf{W}}$ give rise to different multivariate spatial regression models. MCMC model fitting proceeds via a Gibbs sampler with Metropolis steps (see e.g., Carlin and Louis 2000, p. 159) on the marginalized scale, after integrating out $\tilde{\mathbf{W}}$, to reduce the parameter space. In the marginalized model, the response vector is distributed as,

$$\mathbf{Y} \sim MVN(\mathbf{X}\boldsymbol{\beta},\ \mathscr{A}\Sigma_{\tilde{\mathbf{W}}}\mathscr{A}^\top + I_n \otimes \Psi). \tag{7}$$

Bayesian hierarchical models are completed by assigning prior distributions on the parameters. Customarily, we let $\boldsymbol{\beta} \sim MVN(\boldsymbol{\mu}_{\boldsymbol{\beta}}, \Sigma_{\boldsymbol{\beta}})$, a p dimensional multivariate normal distribution. The measurement error dispersion Ψ could be assigned an inverse–Wishart prior, although one usually assumes independence of measurement error for the different response measurements in each site and thus sets $\Psi = diag[\tau_i^2]_{i=1}^m$ as a diagonal matrix. Also recall that $\mathscr{A}$ itself is unknown and needs to be stochastically specified. As mentioned in Section 2, the specific form of $\mathscr{A}$ will depend upon the exact form of $\mathbf{A}$. For the stationary setting, we have $\mathscr{A} = I_n \otimes \mathbf{A}$ and we assign an inverse-Wishart prior to $\mathbf{A}\mathbf{A}^\top$.

Finally, recall that $\Sigma_{\tilde{\mathbf{W}}} = [\tilde{\mathbf{K}}(\mathbf{s}_i - \mathbf{s}_j; \boldsymbol{\theta})]_{i,j=1}^n$ and one needs to assign priors on $\boldsymbol{\theta} = \{\phi_k, \nu_k\}_{k=1}^m$. This will again depend upon the specific choice of the correlation functions. In general the spatial decay parameters are weakly identifiable and prior selection becomes an even more delicate issue, with reasonably informative priors needed for satisfactory MCMC behavior. Typically we set prior distributions for the decay parameters relative to the size of their domains; for instance, by setting the prior means to values that imply spatial ranges of approximately a certain fraction of the maximum intersite distance. For the Matérn correlation function, the smoothness parameter ν is often estimated using a uniform prior distribution with support on the interval (0, 2). This choice is motivated by earlier findings (e.g., Stein 1999) that it is almost impossible for the data to distinguish between these smoothness parameters for values greater than 2.

The set of parameters that are to be updated in the marginalized model from (6) are generically denoting by $\Omega = (\boldsymbol{\beta}, \mathscr{A}, \boldsymbol{\theta}, \Psi)$ with posterior distribution sampled from

$$P(\Omega \,|\, Data) \propto P(\boldsymbol{\beta})P(\mathscr{A})P(\boldsymbol{\theta})P(\Psi)P(\mathbf{Y} \,|\, \boldsymbol{\beta}, \mathscr{A}, \boldsymbol{\theta}, \Psi). \tag{8}$$

An efficient MCMC algorithm is obtained by updating $\boldsymbol{\beta}$ from its $MVN(\boldsymbol{\mu}_{\boldsymbol{\beta}|\cdot}, \Sigma_{\boldsymbol{\beta}|\cdot})$ full conditional, where

$$\begin{aligned} \Sigma_{\boldsymbol{\beta}|\cdot} &= [\Sigma_{\boldsymbol{\beta}}^{-1} + \mathbf{X}^\top(\mathscr{A}\Sigma_{\tilde{\mathbf{W}}}\mathscr{A}^\top + I_n \otimes \Psi)^{-1}\mathbf{X}]^{-1} \\ \text{and } \boldsymbol{\mu}_{\boldsymbol{\beta}|\cdot} &= \Sigma_{\boldsymbol{\beta}|\cdot}\mathbf{X}^\top(\mathscr{A}\Sigma_{\tilde{\mathbf{W}}}\mathscr{A}^\top + I_n \otimes \Psi)^{-1}\mathbf{Y}. \end{aligned} \tag{9}$$

All the remaining parameters have to be updated using Metropolis steps. Depending upon the application, this may be implemented using block-updates (e.g., separate multivariate proposals and subsequent acceptance or rejections for parameters in Ψ, $\mathbf{A}$, and $\boldsymbol{\theta}$). On convergence, the MCMC output generates L samples, say $\{\Omega^{(l)}\}_{l=1}^L$, from the posterior distribution in (8).

3.2. Posterior predictive inference

In updating Ω using the marginal model as outlined above, we do not directly sample the spatial coefficients $\tilde{\mathbf{W}}$ and hence cannot directly obtain $\mathbf{W} = \mathscr{A}\tilde{\mathbf{W}}$. This shrinks the parameter space, resulting in a more efficient MCMC algorithm. A primary advantage of the Gaussian likelihood (as in (6)) is that the posterior distribution of $\tilde{\mathbf{W}}$ can be recovered in a posterior predictive fashion by sampling from

$$P(\tilde{\mathbf{W}}|\ Data) \propto \int P(\tilde{\mathbf{W}}|\Omega,\ Data)P(\Omega|\ Data)d\Omega. \tag{10}$$

Once we have the posterior samples from $P(\Omega|\ Data)$, $\{\Omega^{(l)}\}_{l=1}^{L}$,posterior samples from $P(\tilde{\mathbf{W}}|\ Data)$ drawn by sampling $\tilde{\mathbf{W}}^{(l)}$ for each $\Omega^{(l)}$ from $P(\tilde{\mathbf{W}}\,|\,\Omega^{(l)},\ Data)$. This composition sampling is routine because $P(\tilde{\mathbf{W}}\,|\,\Omega,\ Data)$ in (10) is Gaussian; in fact, from (6) we have this distribution as

$$MVN\left[(\Sigma_{\tilde{\mathbf{W}}}^{-1} + \mathscr{A}^{\top}(I_n \otimes \Psi^{-1})\mathscr{A})^{-1}\mathscr{A}^{\top}(I_n \otimes \Psi^{-1})(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}),\ (\Sigma_{\tilde{\mathbf{W}}}^{-1} + \mathscr{A}^{\top}(I_n \otimes \Psi^{-1})\mathscr{A})^{-1}\right].$$

The posterior estimates of these realizations can subsequently be mapped with contours to produce image and contour plots of the spatial processes.

Next, let $\{\mathbf{s}_{0i}\}_{i=1}^{n^*}$ be a collection of n^* locations where we seek to predict the responses. It might also be of interest to compute the posterior predictive distribution $P(\tilde{\mathbf{W}}^*\,|\,Data)$ where $\tilde{\mathbf{W}}^* = [\tilde{\mathbf{W}}(\mathbf{s}_{0k})]_{k=1}^{n^*}$. Note that

$$P(\tilde{\mathbf{W}}^*|\ Data) \propto \int P(\tilde{\mathbf{W}}^*|\tilde{\mathbf{W}},\Omega,\ Data)P(\tilde{\mathbf{W}}|\Omega,\ Data)P(\Omega|\ Data)d\Omega d\tilde{\mathbf{W}}. \tag{11}$$

This can be computed by composition sampling by first obtaining the posterior samples $\{\Omega^{(l)}\}_{l=1}^{L} \sim P(\Omega|\ Data)$, then drawing $\tilde{\mathbf{W}}^{(l)} \sim P(\tilde{\mathbf{W}}|\Omega^{(l)},\ Data)$ for each l as described in (10) and finally drawing $\tilde{\mathbf{W}}^{*(l)} \sim P(\tilde{\mathbf{W}}^*|\tilde{\mathbf{W}}^{(l)},\ \Omega^{(l)},\ Data)$. This last distribution is derived as a conditional multivariate normal distribution, namely:

$$\begin{pmatrix} \tilde{\mathbf{W}} \\ \tilde{\mathbf{W}}^* \end{pmatrix} \sim MVN\left(\begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \begin{pmatrix} \Sigma_{\tilde{\mathbf{W}}} & \Sigma_{\tilde{\mathbf{W}},\tilde{\mathbf{W}}^*} \\ \Sigma_{\tilde{\mathbf{W}}^*,\tilde{\mathbf{W}}} & \Sigma_{\tilde{\mathbf{W}}^*} \end{pmatrix}\right),$$

$$\text{where } \Sigma_{\tilde{\mathbf{W}}} = [\oplus_{k=1}^{m}\rho_k(\mathbf{s}_i,\mathbf{s}_j;\boldsymbol{\theta}_k)]_{i,j=1}^{n},\ \Sigma_{\tilde{\mathbf{W}}^*} = [\oplus_{k=1}^{m}\rho_k(\mathbf{s}_{0i},\mathbf{s}_{0j};\boldsymbol{\theta}_k)]_{i,j=1}^{n^*},$$
$$\text{and } \Sigma_{\mathbf{W},\mathbf{W}^*}^{\top} = \Sigma_{\tilde{\mathbf{W}}^*,\tilde{\mathbf{W}}} = [\oplus_{k=1}^{m}\rho_k(\mathbf{s}_{0i},\mathbf{s}_j;\boldsymbol{\theta}_k)]_{i=1,j=1}^{n^*,n}.$$

Therefore, the distribution $P(\tilde{\mathbf{W}}^*|\tilde{\mathbf{W}},\Omega,\ Data)$ is $MVN(\boldsymbol{\mu}_{\tilde{\mathbf{W}}^*|\tilde{\mathbf{W}}},\Sigma_{\tilde{\mathbf{W}}^*|\tilde{\mathbf{W}}})$, where

$$\boldsymbol{\mu}_{\tilde{\mathbf{W}}^*|\tilde{\mathbf{W}}} = \Sigma_{\tilde{\mathbf{W}},\tilde{\mathbf{W}}^*}^{\top}\Sigma_{\tilde{\mathbf{W}}}^{-1}\tilde{\mathbf{W}} \text{ and } \Sigma_{\tilde{\mathbf{W}}^*|\tilde{\mathbf{W}}} = \Sigma_{\tilde{\mathbf{W}}^*} - \Sigma_{\tilde{\mathbf{W}},\tilde{\mathbf{W}}^*}^{\top}\Sigma_{\tilde{\mathbf{W}}}^{-1}\Sigma_{\tilde{\mathbf{W}},\tilde{\mathbf{W}}^*}.$$

Once $\{\tilde{\mathbf{W}}^{*(l)}\}_{l=1}^{L}$ have been obtained, we can easily predict the responses, say $\mathbf{Y}^* = [\mathbf{Y}(\mathbf{s}_{0i})]_{i=1}^{n^*}$ at those sites as long as the $mn^* \times p$ matrix of regressors for those locations, say $\mathbf{X}^*$, is available. This can be done by simply sampling the conditional expectations $E[\mathbf{Y}^*|\ Data]^{(l)} = \mathbf{X}^*\boldsymbol{\beta}^{(l)} + \mathscr{A}^{(l)}\tilde{\mathbf{W}}^{*l}$ for $l = 1,\ldots,L$. Equivalently, predictions can be executed by drawing posterior samples from the marginal distribution below, without resorting to direct updates of the $\tilde{\mathbf{W}}$ as follows:

$$P(\mathbf{Y}^*|\ Data) \propto \int P(\mathbf{Y}^*|\Omega,\ Data)P(\Omega|\ Data)d\Omega. \tag{12}$$

In the stationary setting with the marginalized model, observe that

$$\begin{pmatrix} \mathbf{Y} \\ \mathbf{Y}^* \end{pmatrix} \sim MVN\left(\begin{pmatrix} \mathbf{X}\boldsymbol{\beta} \\ \mathbf{X}^*\boldsymbol{\beta} \end{pmatrix}, \begin{pmatrix} \Sigma_{\mathbf{Y},\mathbf{Y}} & \Sigma_{\mathbf{Y},\mathbf{Y}^*} \\ \Sigma_{\mathbf{Y},\mathbf{Y}^*}^\top & \Sigma_{\mathbf{Y}^*,\mathbf{Y}^*} \end{pmatrix}\right),$$

$$\begin{aligned} \text{where } \Sigma_{\mathbf{Y},\mathbf{Y}} &= \mathscr{A}\Sigma_{\tilde{\mathbf{W}}}\mathscr{A}^\top + I_n \otimes \Psi \text{ with } \mathscr{A} = I_n \otimes \mathbf{A}, \\ \Sigma_{\mathbf{Y}^*,\mathbf{Y}^*} &= \mathscr{A}^*\Sigma_{\tilde{\mathbf{W}}^*}\mathscr{A}^{*T} \text{ with } \mathscr{A}^* = (I_{n^*} \otimes \mathbf{A}), \\ \text{and } \Sigma_{\mathbf{Y},\mathbf{Y}^*}^\top &= \mathscr{A}^*\Sigma_{\tilde{\mathbf{W}}^*,\tilde{\mathbf{W}}}\mathscr{A}^\top. \end{aligned}$$

Therefore, the distribution $P(\mathbf{Y}^*|\Omega,\ Data)$ is $MVN(\boldsymbol{\mu}_{\mathbf{Y}^*|\mathbf{Y}}, \Sigma_{\mathbf{Y}^*|\mathbf{Y}})$, where

$$\boldsymbol{\mu}_{\mathbf{Y}^*|\mathbf{Y}} = \mathbf{X}^*\boldsymbol{\beta} + \Sigma_{\mathbf{Y},\mathbf{Y}^*}^\top \Sigma_{\mathbf{Y},\mathbf{Y}}^{-1}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) \text{ and } \Sigma_{\mathbf{Y}^*|\mathbf{Y}} = \Sigma_{\mathbf{Y}^*,\mathbf{Y}^*} - \Sigma_{\mathbf{Y},\mathbf{Y}^*}^\top \Sigma_{\mathbf{Y},\mathbf{Y}}^{-1}\Sigma_{\mathbf{Y},\mathbf{Y}^*}.$$

Simulating from $P(\mathbf{Y}^*|\Omega,\ Data)$ is routine for any given Ω. Hence, the predictive distribution is again obtained using composition sampling: for each $\Omega^{(l)} \sim P(\Omega\,|\,Data)$, we draw $\mathbf{Y}^{*,(l)} \sim P(\mathbf{Y}^*|\Omega^{(l)},\ Data)$ to obtain posterior predictive samples $\{\mathbf{Y}^{*,l}\}_{l=1}^L$.

3.3. Model selection

Since we consider several alternative models with varying degrees of spatial richness, we use the Deviance Information Criterion (DIC, Spiegelhalter *et al.* 2002) as a measure of model choice. The DIC has nice properties for Gaussian likelihoods such as ours, and is particularly convenient to compute from posterior samples. This criterion is the sum of the Bayesian deviance (a measure of model fit), and the effective number of parameters (a penalty for model complexity). Lower DIC values indicate the preferred models. The deviance, up to an additive quantity not depending upon Ω, is simply the negative of twice the log-likelihood, $D(\Omega) = -2\log L(Data|\ \Omega)$, where $L(Data|\ \Omega)$ is the first stage Gaussian likelihood from (6) for the respective models. The Bayesian deviance is the posterior mean, $\overline{D(\Omega)} = E_{\Omega|\mathbf{Y}}[D(\Omega)]$, while the effective number of parameters is given by $p_D = \overline{D(\Omega)} - D(\bar{\Omega})$, where $\bar{\Omega}$ is the posterior mean of the model parameters Ω. The DIC is then given by $\overline{D(\Omega)} + p_D$.

DIC and associated statistics can be calculated from either the unmarginalized (6) or marginalized (7) likelihood. In addition to lower DIC scores, shrinkage among the random spatial effects might be considered a metric for improved model fit. This shrinkage can be seen in reduced p_D values calculated with the unmarginalized model.

4. Illustrating spBayes

4.1. spBayes

spBayes is an R package that currently hosts two core functions, `ggt.sp` and `sp.predict`, which implement the methods presented in Section 3. The underlying MCMC sampling and associated matrix algebra is written in C++ and leverages **BLAS**, **LAPACK**, and **LINPACK** FORTRAN subroutines. **spBayes** was written specifically as an R extension; therefore, lower–level calls to C++ and FORTRAN functions are through R's foreign language interfaces which allow portability across operating systems.

Because **spBayes** relies on lower–level calls to **BLAS** and **LAPACK** routines, users can realize substantial gains in sampling efficiency by using processor optimized versions of these packages. Depending on the operating system and processor, R can interface with several efficient **BLAS** and **LAPACK** implementations including: AMD Core Math Library (ACML); Math Kernel Library (MKL); Goto BLAS; and Automatically Tuned Linear Algebra Software (ATLAS).

The `ggt.sp` and supporting functions accommodate any number of observations, response variables, and associated covariates. Computing time is the only restriction. The principal factor that dictates computing time is the inversion of the $mn \times mn$ covariance matrix $\Sigma_{\mathbf{W}}(\boldsymbol{\theta}) + I_n \otimes \Psi$ that is required for each MCMC iteration. Within **spBayes**, this inversion is accomplished with **LAPACK**'s `dpotrf` and `dpotri` routines (i.e., a Cholesky decomposition and subsequent inversion). For example, on a modern single processor computer and using R's internal non-optimized **BLAS** and **LAPACK**, `ggt.sp` can generate about 5,000 posterior samples within an hour given a model with $mn = 500$. As noted above, processor optimized **BLAS** and **LAPACK** can significantly increase the speed of these routines.

4.2. Synthetic data

A synthetic data set serves to validate our proposed modeling approach and to demonstrate the use of DIC to assess model fit. The synthetic data set describes a stationary, isotropic, non–separable multivariate process. For simplicity, we consider only two response variables (i.e., $m = 2$). The Matérn correlation function (2) with $\nu = 0.5$ was used to produce the data's spatial dependence structure. Fixing ν at 0.5 reduces the Matérn to the familiar exponential correlation function, $\rho(\mathbf{s} - \mathbf{s}'; \phi) = \exp(-\phi \|\mathbf{s} - \mathbf{s}'\|)$. Thus we take $\tilde{\mathbf{K}}(\mathbf{s} - \mathbf{s}'; \boldsymbol{\theta}) = diag[\rho_i(\mathbf{s} - \mathbf{s}'; \phi_i)]_{i=1}^2$ where $\boldsymbol{\theta} = (\phi_1, \phi_2)$. The multivariate process was generated with the following parameters:

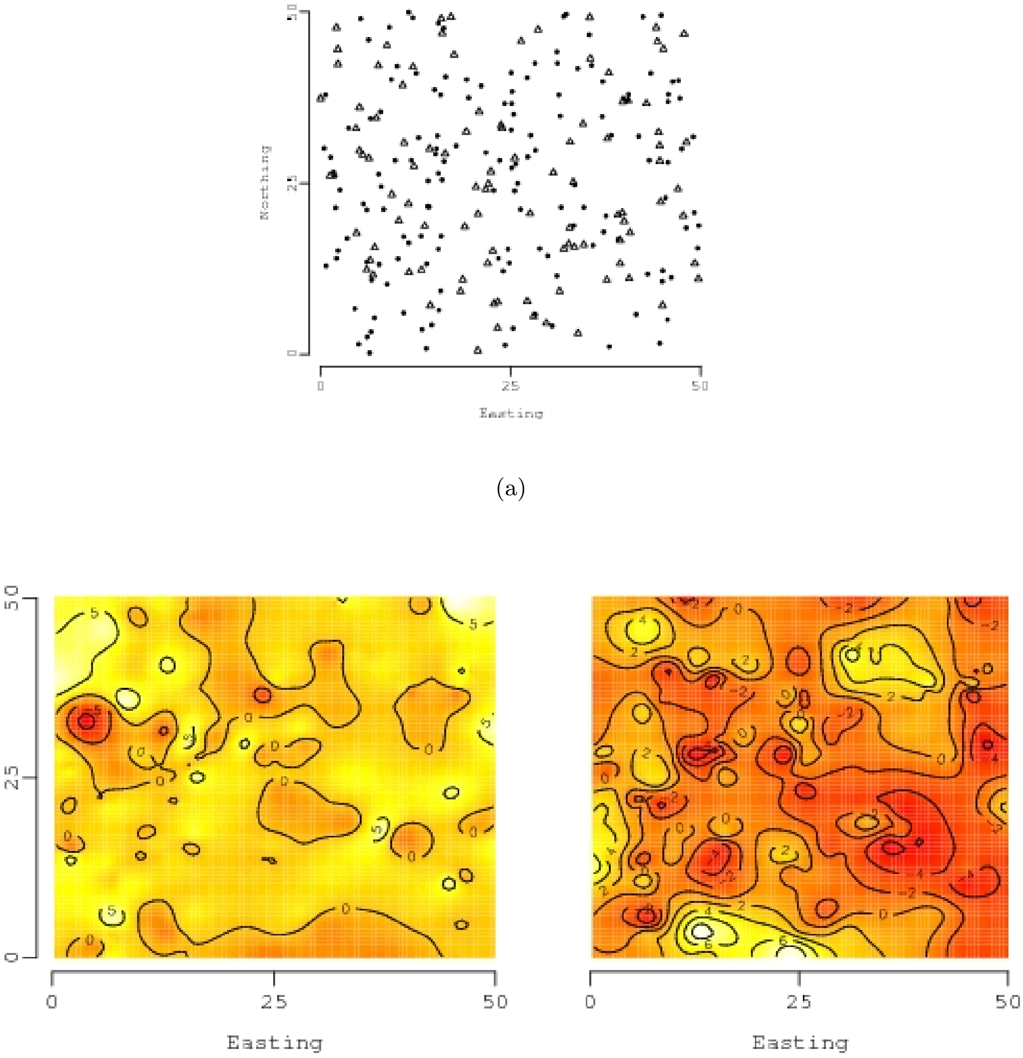
$$\boldsymbol{\beta} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \ \mathbf{K}(\mathbf{0}; \boldsymbol{\theta}) = \begin{pmatrix} 1 & -2 \\ -2 & 8 \end{pmatrix}, \ \Psi = \begin{pmatrix} 9 & 0 \\ 0 & 2 \end{pmatrix}, \ \boldsymbol{\phi} = \begin{pmatrix} 0.6 \\ 0.1 \end{pmatrix}.$$

This yields $\mathbf{A} = \mathbf{K}^{1/2}(\mathbf{0}; \boldsymbol{\theta}) = \begin{pmatrix} 1 & 0 \\ -2 & 2 \end{pmatrix}$. The above specifications describe a multivariate process with independent non–spatial variance among the response surfaces and a strong negative cross–correlation between the spatial processes (-0.707 to be precise).

The effective range of spatial dependence (i.e., the distance at which the correlation drops to 0.05) is determined by $-log(0.05)/\phi$. The range parameters in $\boldsymbol{\phi}$ provide an effective range of ~ 5.0 units for the first response surface and ~ 30.0 units for the second. Given these parameters and the marginalized likelihood (7), we can draw realizations of the desired co–varying response surfaces. Given the 150 randomly selected locations (•) in Figure 1(a), Figures 1(b) and 1(c) offer one such realization of these spatial processes. Univariate empirical semivariograms for this synthetic data along with restricted maximum likelihood (REML) estimates for the exponential covariance function parameters are provided in Figure 2.

4.3. Inference and model choice using `ggt.sp`

The generalized template introduced in Section 3 suggests several potential models. Here we consider seven stationary process models of increasing complexity. Our focus is on the alternative specifications of $\mathscr{A}$ and $\tilde{\mathbf{W}}$ within (6). For each model, we assume an isotropic spatial process that can be modeled with the exponential correlation function.



(a)

(b)

(c)

Figure 1: (a) Random points used to generate the synthetic data set. Subsequent parameter estimation is based on 150 (•) locations, and prediction at the 100 open triangle locations (△). Plots (b) and (c) are interpolated surfaces of the first and second response variables, respectivly, generated with the random points in (a), given parameters, and the marginalized likelihood 7.

A simple linear regression model (no random effects) is

$$\text{Model 1: } \mathscr{A}\tilde{\mathbf{W}} = \mathbf{0}.$$

This model would suffice in the presence of negligible extraneous variation beyond what is explained by the model's regressors. However, if autocorrelation is seen in each response variable's surface, as in Figure 1, and the regressors do not account for this association as a

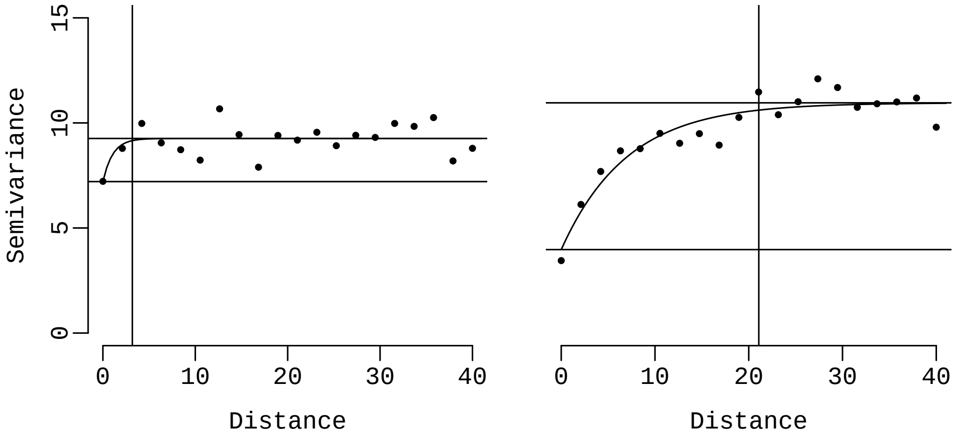


Figure 2: Empirical semivariograms and exponential function REML parameter estimates for the two synthetic response surfaces. The estimates of the nugget (bottom horizontal line), sill (upper horizontal line), and range (vertical line) for the first surface are approximately 7.2, 9.3, and 3.2 respectively. The corresponding semivariogram estimates of nugget, sill, and range for the second response surface (right plot) are 4.0, 11.0, and 21.0, respectively.

function of distance between locations, then this model violates the implicit assumption of conditionally independent observations.

The next four spatial models impose separable association structures as in (5). For each model, $\Sigma_{\tilde{\mathbf{W}}} = [\tilde{\mathbf{K}}(\mathbf{s}_i - \mathbf{s}_j; \boldsymbol{\phi})]_{i,j=1}^n$, $\boldsymbol{\phi} = \{\phi\}_{k=1}^m$ implies the response variables share a common spatial decay parameter. The first, and simplest, of these models assumes common spatial variance (i.e., σ^2) and no non–spatial variance,

$$\text{Model 2: } \mathbf{A} = \sigma I_m \text{ and } \Psi = \mathbf{0}.$$

The second spatial model allows for a common pure error variance term (i.e., τ^2),

$$\text{Model 3: } \mathbf{A} = \sigma I_m \text{ and } \Psi = \tau^2 I_m.$$

The next model extends Model 3 to allow response specific spatial and pure error variance terms,

$$\text{Model 4: } \mathbf{A} = diag[\sigma_i]_{i=1}^m \text{ and } \Psi = diag[\tau_i^2]_{i=1}^m.$$

Where Model 4 assumes independence among the response surfaces' spatial variance, Model 5 explicitly models the off–diagonal element in the cross–covariance matrix $\mathbf{K}$,

$$\text{Model 5: } \mathbf{A} \text{ and } \Psi = diag[\tau_i^2]_{i=1}^m$$

where, recall, $\mathbf{A}$ is the square root of the $m \times m$ cross–covariance matrix. The sixth model is the non–separable form of Model 5, allowing response specific spatial range terms,

$$\text{Model 6: } \mathbf{A}, \Psi = diag[\tau_i^2]_{i=1}^m, \text{ and } \boldsymbol{\phi} = \{\phi_k\}_{k=1}^m$$

The final candidate model, extends the diagonal pure error cross–covariance matrix in Model 6 to a full matrix, allowing for non–spatial variance dependence among the response surfaces,

$$\text{Model 7: } \mathbf{A}, \Psi, \text{ and } \boldsymbol{\phi} = \{\phi_k\}_{k=1}^{m}.$$

The `ggt.sp` function within **spBayes** was used to fit each candidate model to the synthetic data. Following the methods described in Section 3.1, the `ggt.sp` function generates samples from each parameter's posterior distribution. The following code blocks detail the `ggt.sp` arguments used to fit Model 6; the other candidate models can be fit using variations on these specifications.

The arguments for the `ggt.sp` function follow a logical progression through a Bayesian modeling process. The initial step is to assign appropriate prior distributions to each parameter. The prior distributions used in `ggt.sp` are consistent with definitions found in Appendix A of Gelman *et al.* (2004). The variance parameters in Model 6 are a $m \times m$ spatial cross-covariance matrix $\mathbf{K}$, diagonal non–spatial cross–covariance matrix Ψ, and m spatial range parameters $\boldsymbol{\phi}$. The code block below specifies an inverse–Wishart prior for $\mathbf{K}$, separate inverse–Gamma priors for the diagonal elements of Ψ, and a single uniform prior which will be used for both of the ϕ. The empirical semivariograms (Figure 2) were used to help define the priors' hyperparameters. For instance, the semivariograms suggest partial sills of about 3 and 6 for the two conditional response surfaces (i.e., conditional on the regressors), and therefore these values serve as the diagonal elements in the $m \times m$ shape hyperparameter of the inverse–Wishart prior used for $\mathbf{K}$. The nugget estimates from the semivariograms can guide the choice of the inverse–Gamma scale hyperparameters (i.e., 7 and 5, respectively). A shape of 2 in the inverse–Gamma suggests a distribution with infinite variance centered on the scale hyperparameter. An alternative to the inverse–Gamma is the half–Cauchy also available in **spBayes** (see Gelman 2006, for further discussion). The `LOGUNIF` prior on $\boldsymbol{\phi}$ indicates a uniform distribution with support strictly greater than zero. As noted at the end of Section 4.2, $\sim 3/\phi$ can be considered the effective spatial range. Therefore, a vague prior on $\boldsymbol{\phi}$ is a uniform distribution with support on the interval (3/3, 3/0.06) or (1, 50) distance units.

```
K.prior <- prior(dist="IWISH", df=2, S=diag(c(3, 6)))
Psi.prior.1 <- prior(dist="IG", shape=2, scale=7)
Psi.prior.2 <- prior(dist="IG", shape=2, scale=5)
phi.prior <- prior(dist="LOGUNIF", a=0.06, b=3)
```

The next portion of code describes how each variance parameter is updated, the assigned prior, starting values, and the order in which blocks of parameters are updated. The variance parameters within the proposed model template are updated with the Metropolis–Hastings algorithm. This algorithm requires a *tuning* or *step–size* value which defines the variance term in the multivariate normal proposal density. A tuning value or matrix of the appropriate dimension must be assigned to each variance parameter. If a matrix tuning value is specified, it must be invertible. Within the R list data structure below, the element tags (e.g., `K`, `Psi`, `phi`, and if the Matérn correlation function is used, `nu`) are keywords used by `ggt.sp` to link the directives with the given model parameter. Below, for example, the identity matrix serves as the starting value for $\mathbf{K}$. Only three elements need to be updated in the symmetric $\mathbf{K}$ matrix; therefore, the tuning matrix is 3×3 with diagonal elements assigned to $\mathbf{K}$'s lower–triangle column major elements (i.e., the tuning values are $\mathbf{K}_{1,1} = 0.1$, $\mathbf{K}_{2,1} = 0.5$, $\mathbf{K}_{2,2} = 0.1$).

Updating parameters in blocks often provides finer control on Metropolis–Hastings acceptance rate. The value assigned to the `sample.order` list tag controls the block update sequence. The code below describes a sequential Metropolis–Hastings sampling scheme that first accepts/rejects the proposed $\mathbf{K}$ given the currently accepted Ψ and $\boldsymbol{\phi}$ samples, then accepts/rejects Ψ given the currently accepted $\mathbf{K}$ and $\boldsymbol{\phi}$, then accepts/rejects $\boldsymbol{\phi}$ given the currently accepted $\mathbf{K}$ and Ψ.

```
var.update.control <-
   list("K"=list(sample.order=0, starting=diag(1, 2),
          tuning=diag(c(0.1, 0.5, 0.1)), prior=K.prior),

        "Psi"=list(sample.order=1, starting=1,
          tuning=0.3, prior=list(Psi.prior.1, Psi.prior.2)),

        "phi"=list(sample.order=2, starting=0.5,
          tuning=0.5, prior=list(phi.prior, phi.prior))
        )
```

In the `var.update.control` list, the data structure assigned to the `prior` tags determines several model characteristics. Model 6 is non–separable; therefore, the `prior` list within the `phi` tag defines two priors, corresponding to the first and second spatial process. If, as in the `phi` and `Psi` lists above, multiple priors are defined, the `starting` and `tuning` values are recycled. Alternatively, starting value and tuning vectors can be passed in place of the scalars.

The regression parameters, $\boldsymbol{\beta}$, also receive a prior and update method. Unlike the variance parameters, the $\boldsymbol{\beta}$ are updated using either Gibbs (9) or Metropolis–Hastings. The code below specifies a flat prior with Gibbs updating for each element in the $\boldsymbol{\beta}$ parameter vector.

```
beta.control <- list(update="GIBBS", prior=prior(dist="FLAT"))
```

The last argument that is passed to `ggt.sp` defines the number of posterior samples to collect and, if random spatial effects, $\mathscr{A}\tilde{\mathbf{W}}$, should be recovered.

```
run.control <-
  list("n.samples"=5000, "sp.effects"=TRUE)
```

These arguments are then passed to `ggt.sp` along with additional model specifications and data as given in the code below.

```
ggt.Model.6 <- ggt.sp(formula=list(Y.1~1, Y.2~1),
                      run.control=run.control,
                      coords=coords,
                      var.update.control=var.update.control,
                      beta.update.control=beta.control,
                      cov.model="exponential")
```

Specifically, the `ggt.sp` call includes: a list of m symbolic model statements `formula=list(Y.1∼1, Y.2∼1)` along with an optional `data` argument similar to R's `lm()`

function; a `coords` argument which specifies the $n \times 2$ matrix of observation coordinates; and the covariance model, `cov.model`, to be used (i.e., `exponential`, `matern`, `spherical`, or `gaussian`). The return object, `ggt.Model.6` includes posterior samples for each parameter and random spatial effect samples.

Finally, as described in Section 3.3, a call to `sp.DIC` provides both the marginalized and unmarginalized DIC for the given `ggt.sp` return object. The code below calculates DIC over the interval of 1,000 to 5,000 samples. The `sp.DIC` function also accepts `thin` and `end` to control the sample interval over which DIC is calculated.

```
sp.DIC(ggt.Model.6, DIC.marg=TRUE, DIC.unmarg=TRUE, start=1000)
```

4.4. Synthetic data analysis results

We fit the seven competing models to the synthetic data. For each of these models, three MCMC chains were run for 5,000 iterations. Chain mixing occurred within 1,000 iterations, therefore the remaining 12,000 samples (4,000×3) were retained for posterior analysis.

DIC was used to select the best candidate model to produce response specific surfaces of random spatial effects $E[\mathbf{W}|\ Data]$, and subsequent predictions $E[\mathbf{W}^* \mid Data]$ and $E[\mathbf{Y}^*|\ Data]$.

Table 1 provides p_D and DIC scores for the candidate models. As noted in Section 3.3, lower DIC scores suggest better model performance; therefore, Models 5, 6, and 7 are preferred. Based on the actual number of parameters, Model 5 is the most parsimonious of the three. Separability is the distinction between Model 5 and 6. Even though Model 6 gave rise to the data, it is common that the notoriously ill–defined ϕ does not contribute much to model distinctions in formal model fit comparisons (e.g., DIC). Rather, we might look to the interpolated residual surfaces, empirical semivariograms, and $\boldsymbol{\phi}$ estimates to determine if there is an advantage to the non–separable model. In this case, there is a strong distinction in spatial dependence trends in Figure 1 surfaces and Figure 2, and between estimates of ϕ_1 and ϕ_2 in Table 2. Following these diagnostics, we select Model 6 over Model 5.

The remaining choice is between Model 6 and 7. Unlike Model 6, Model 7 explicitly models the off–diagonal element in Ψ. The unmarginalized DIC score in Table 1 suggests that Model 7 might be the best fit. However, the marginalized DIC score for Model 7 does not show an advantage over Model 6. To better understand the contribution of the off–diagonal elements in Ψ or $\mathbf{K}$, it is often useful to convert these covariance matrices to the corresponding correlation matrices. For Model 7, the bounds of the 95% credible interval about Ψ's off–diagonal correlation element are -0.749 and 0.656, which suggests that the non–spatial correlation between the response surfaces is not significant. Therefore, Model 7 does not seem to add additional insight or a consistently superior model DIC score over Model 6.

Under models with $\Psi = \mathbf{0}$, for instance Models 1 and 2, the unmarginalized target likelihood reduces to the marginalized likelihood. Therefore, the unmarginalized DIC and associated statistics are omitted from Table 1.

Table 2 provides parameter estimate summaries for Model 6. Because the synthetic data is only one realization of the defined model, it would be a rare occurrence for the *true* parameters to fall outside of the estimated 2.5%–97.5% percentiles (i.e., only a 5% chance). Indeed, Table 2 shows that the estimated intervals cover the corresponding true parameters defined in Section 4.2. Further, converting the estimates for the cross–covariance matrix $\mathbf{K}$ to the

		Marginalized		Unmarginalized	
Model	Parameters	p_D	DIC	p_D	DIC
Model 1	τ^2	3.049	993.013	–	–
Model 2	ϕ, σ^2	3.520	987.069	–	–
Model 3	ϕ, σ^2, τ^2	4.515	979.409	233.626	1,643.619
Model 4	ϕ, σ^2_m, τ^2_m	5.382	966.157	256.186	1,537.190
Model 5	ϕ, $\mathbf{A}$, τ^2_m	6.464	958.586	260.640	1,450.184
Model 6	ϕ_m, $\mathbf{A}$, τ^2_m	6.141	957.195	236.838	1,492.053
Model 7	ϕ_m, $\mathbf{A}$, Ψ	5.879	957.882	206.663	1,446.000

Table 1: Synthetic data model comparison using the DIC criterion. For each model both marginalized and unmarginalized p_D and DIC were calculated from three chains of 5,000 samples.

50% (2.5%, 97.5%) percentiles of the posterior cross–correlation matrix, we see that the off–diagonal element is -0.475 (-0.791, -0.095), again covering the true negative association between the response surfaces of -0.707.

Parameter	Estimates: 50% (2.5%, 97.5%)
$\beta_{1,0}$	1.086 (0.555, 1.628)
$\beta_{2,0}$	-0.273 (-1.619, 1.157)
$\mathbf{K}_{1,1}$	1.801 (0.542, 6.949)
$\mathbf{K}_{2,1}$	-1.784 (-3.604, -0.357)
$\mathbf{K}_{2,2}$	8.253 (4.645, 13.221)
$\Psi_{1,1}$	7.478 (3.020, 10.276)
$\Psi_{2,2}$	2.276 (0.832, 5.063)
ϕ_1	1.024 (0.243, 2.805)
ϕ_2	0.193 (0.073, 0.437)

Table 2: Percentiles of the posterior distributions of the parameters in Model 6. β subscripts refer to the response variable and parameter, respectively. Subscripts on $\mathbf{K}$ and Ψ refer to the covariance matrix element. Subscripts on the spatial range parameters, ϕ, refer to the response variable. Summaries generated from three chains of 4,000 samples.

Figure 3 provides the interpolated surfaces of recovered random spatial effects $E[\mathbf{W}|\ Data]$. Conditioning only on the models' intercept terms, that are constant across the domain, causes the mean random effect surfaces to look nearly identical to the observed surfaces Figure 1.

4.5. Prediction using `sp.predict`

The `sp.predict` function will predict $\mathbf{W}^*$ and $\mathbf{Y}^*$ for any set of n^* new sites, given a `ggt.sp` return object and the coordinates and covariates of the new sites. Following the methods detailed in Section 3.2, `sp.predict` returns samples from the joint posterior predictive distribution of the set of new sites. This process is illustrated using the `ggt.sp` return object, `ggt.Model.6`, and the 100 prediction sites in Figure 1(a) (i.e., sites denoted by a triangle symbol). The code below calls the `sp.predict` function.

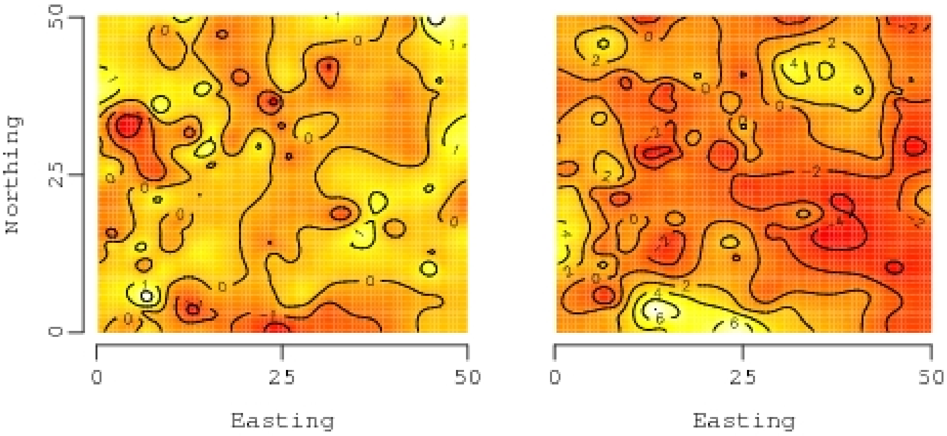


Figure 3: Interpolated surfaces of the recovered random spatial effects for Model 6, $E[\mathbf{W}|\ \mathit{Data}]$.

```
sp.pred <-
  sp.predict(ggt.Model.6, pred.coords=coords, pred.covars=covars)
```

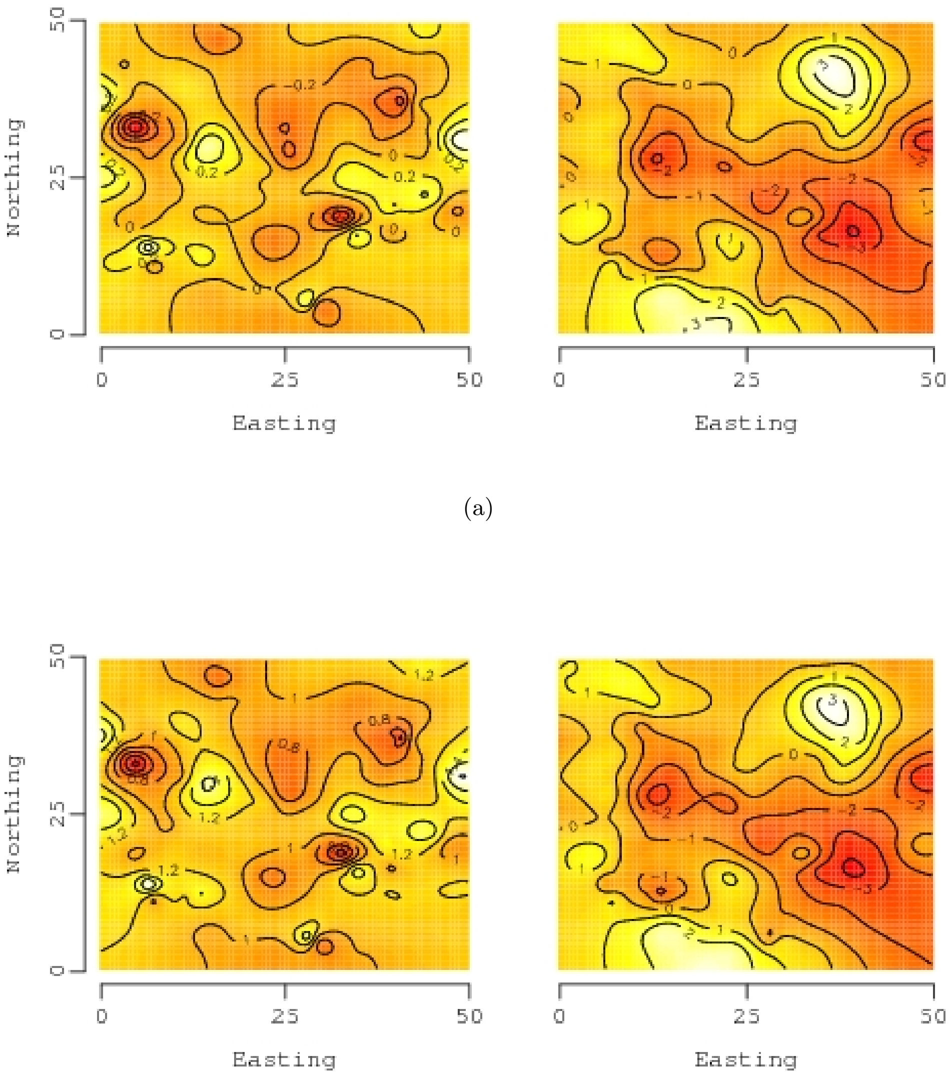
In the above code, the `pred.coords` value is the $n^* \times 2$ matrix of new coordinates and `pred.covars` is the $n^* \times p$ design matrix $\mathbf{X}^*$. This multivariate design matrix can be created by passing a list of the m models' design matrices to the `mk.mv.X` function provided in **spBayes**. The `sp.predict` function also accepts `start`, `thin`, and `end` to control the sample interval over which the predictions are calculated.

Figure 4(a) provides an interpolated surface generated from each new point's mean predicted random spatial effect, $E[\mathbf{W}^* \,|\, \mathit{Data}]$. Without covariates, prediction relies on the strength of the estimated spatial dependence among observed sites and the intercept term which serves as a constant offset across the domain. Both predicted surfaces in Figure 4(a) show substantial smoothing. Smoothing is exacerbated as spatial range increases, as seen in the second response surface predictions. As noted in Table 2, neither intercept term is significant; therefore, Figure 4(b) appears to be only marginally different than the surface of predicted random effects.

4.6. Forest inventory data

Maps of forest attributes are important for quantifying forest carbon dynamics, monitoring forest habitat change, forecasting wood availability, and a host of other forest management and environmental initiatives. The data used in the example below is taken from permanent georeferenced forest inventory plots on the USDA Forest Service Bartlett Experimental Forest (BEF) in Bartlett, New Hampshire. The 1,053 hectare BEF covers a large elevation gradient from the village of Bartlett in the Saco River valley at 207 meters to about 914 meters above sea level. For this illustration, the focus is on predicting the spatial distribution of basal area and total tree biomass per hectare across the BEF.

Basal area is the cross–sectional area of a tree stem 1.37 meters above the ground and is

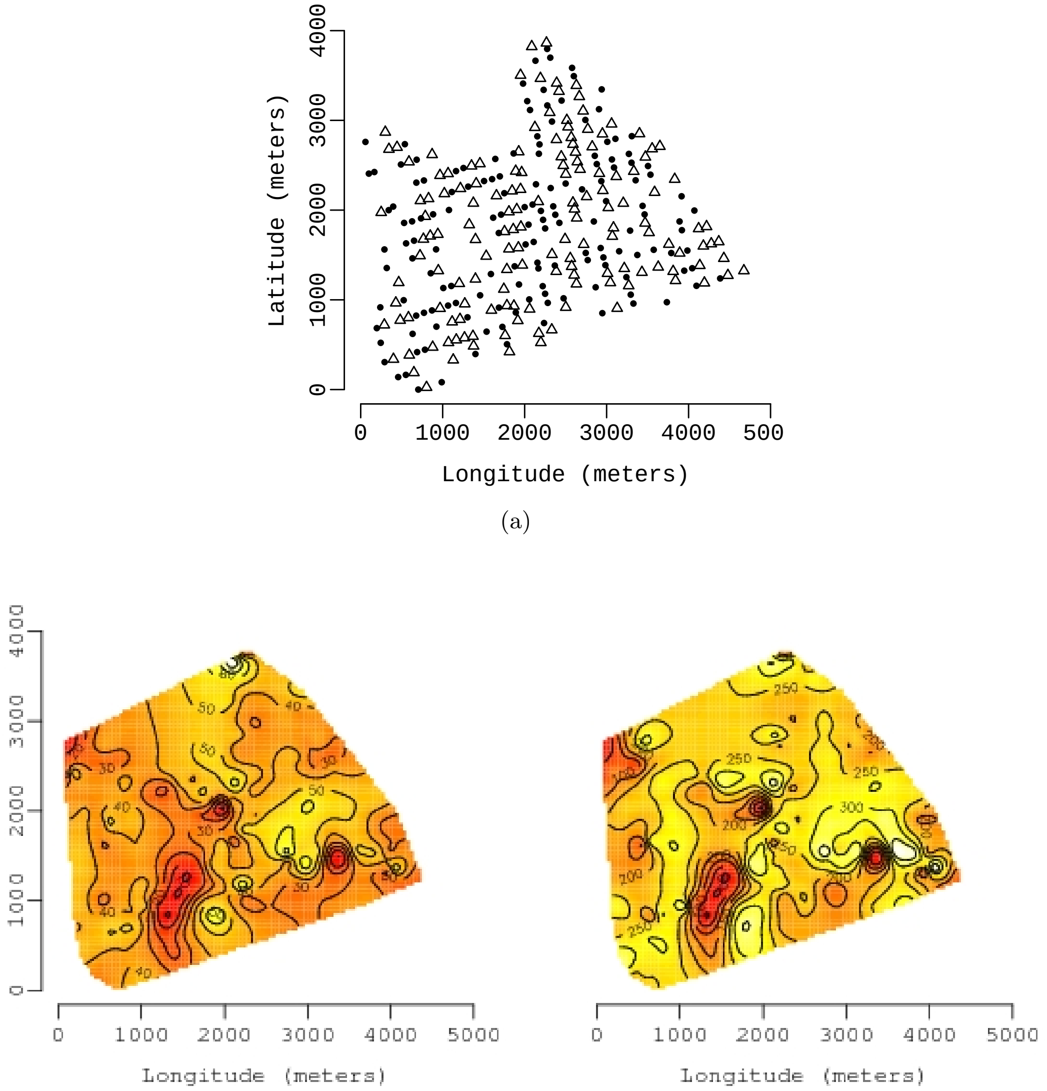


(a)

(b)

Figure 4: (a) Interpolated surfaces of the predicted random spatial effects for Model 6, $E[\mathbf{W}^* \,|\, Data]$. (b) Interpolated surfaces of the posterior predictive distributions for Model 6, $E[\mathbf{Y}^* |\, Data]$.

measured as square meters per hectare. Tree biomass is measured as the weight of all above ground portions of the tree, expressed here as metric tons per hectare. Within the data set, basal area (BA) and biomass (BIO) per hectare are recorded at 300 forest inventory plots. Satellite imagery and other remotely sensed variables have proved useful regressors for predicting these attributes. One spring 2002 date of 30×30 Landsat 7 ETM+ satellite imagery was acquired for the BEF. The image was transformed to tasseled cap components



(a)

(b)

(c)

Figure 5: (a) Forest inventory plots across the Bartlett Experimental Forest. The 300 plots were divided randomly into 150 plots used for parameter estimation denoted with solid dot symbols (●) and the remaining 150 used for prediction marked with triangle symbols (△). Plots (b) and (c) are interpolated surfaces of basal area per hectare and metric tons of biomass per hectare, respectivly.

of brightness (1), greenness (2), and wetness (3) using data reduction techniques. The three resulting spectral variables are labeled TC1, TC2, and TC3. In addition to these spectral variables, digital elevation model data was used to produce a 30×30 elevation (ELEV) layer for the BEF. The centroids of the 300 georeferenced inventory plots were intersected with the elevation (ELEV) and spectral variables.

To demonstrate parameter estimation and prediction, we selected randomly 150 inventory plots for model construction and left the remaining 150 for subsequent predictive mapping. For reference, the 150 model points in Figure 5(a) are used to produce an interpolated surface for each of the two response variables, Figures 5(b) and 5(c).

Previous results suggest that there is positive spatial and non–spatial association between the conditional response surfaces (i.e., conditional on the regressors). Further, the univariate empirical semivariograms for the response variables show some disparity between the spatial ranges, specifically, spatial dependence among sites is smaller for measures of basal area per hectare. Therefore, we fit a non–separable spatial regression with full spatial and non–spatial cross–covariance matrices, $\mathbf{K}$ and Ψ. Further, we assume that spatial dependence can be modeled with the simple exponential correlation function. This specification corresponds to Model 7 in Section 4.3.

As in the previous illustration, the univariate empirical semivariograms provide guidance for starting values and prior hyperparameters. The `ggt.sp` directives for one of the six chains used for model parameter estimation are detailed in the code blocks below. Again, defining priors is the first step in the modeling process.

```
K.Psi.prior <- prior(dist="IWISH", df=2, S=matrix(c(100, 0, 0, 2600), 2, 2))
phi.prior <- prior(dist="LOGUNIF", a=0.0015, b=0.03)
```

The inverse–Wishart prior is used for both cross–covariance matrices. The empirical semivariograms suggest that the nugget and partial sill are about equal for each response variable (i.e., $\sim$ 100 for basal area and $\sim$ 2600 for biomass). The vague scale matrix defined for the common inverse–Wishart centers each response on the suggested values but does not impose any off–diagonal association. The noninformative prior on the spatial range parameters corresponds to an interval of 100 to 2,000 meters.

The following code block defines the starting values of $\mathbf{K}^{1/2}$ and $\Psi^{1/2}$; Metropolis–Hastings tuning values; update method for the $\boldsymbol{\beta}$; the number of samples to take; and the regression model for each response surface.

```
K.Psi.starting <- matrix(c(10, 80, 0, 10), 2, 2)

var.update.control <-
  list("K"=list(sample.order=0, starting=K.Psi.starting,
         tuning=diag(c(0.15, 1.75, 0.15)), prior=K.Psi.prior),
       "Psi"=list(sample.order=1, starting=K.Psi.starting,
         tuning=diag(c(0.15, 1.75, 0.15)), prior=K.Psi.prior),
       "phi"=list(sample.order=2, starting=0.006,
         tuning=0.5, prior=list(phi.prior, phi.prior)))

beta.control <- list(update="GIBBS", prior=prior(dist="FLAT"))

run.control <- list("n.samples"=10000, "sp.effects"=TRUE)

resp.1 <- BA~ELEV+TC1+TC2+TC3
resp.2 <- BIO~ELEV+TC1+TC2+TC3
```

Finally these directives are passed to the `ggt.sp` function along with additional arguments which define the model plot coordinates and the desired spatial correlation function.

```
ggt.chain.1 <-
  ggt.sp(formula=c(resp.1, resp.2), run.control=run.control,
         coords=coords, var.update.control=var.update.control,
         beta.update.control=beta.control,
         cov.model="exponential")
```

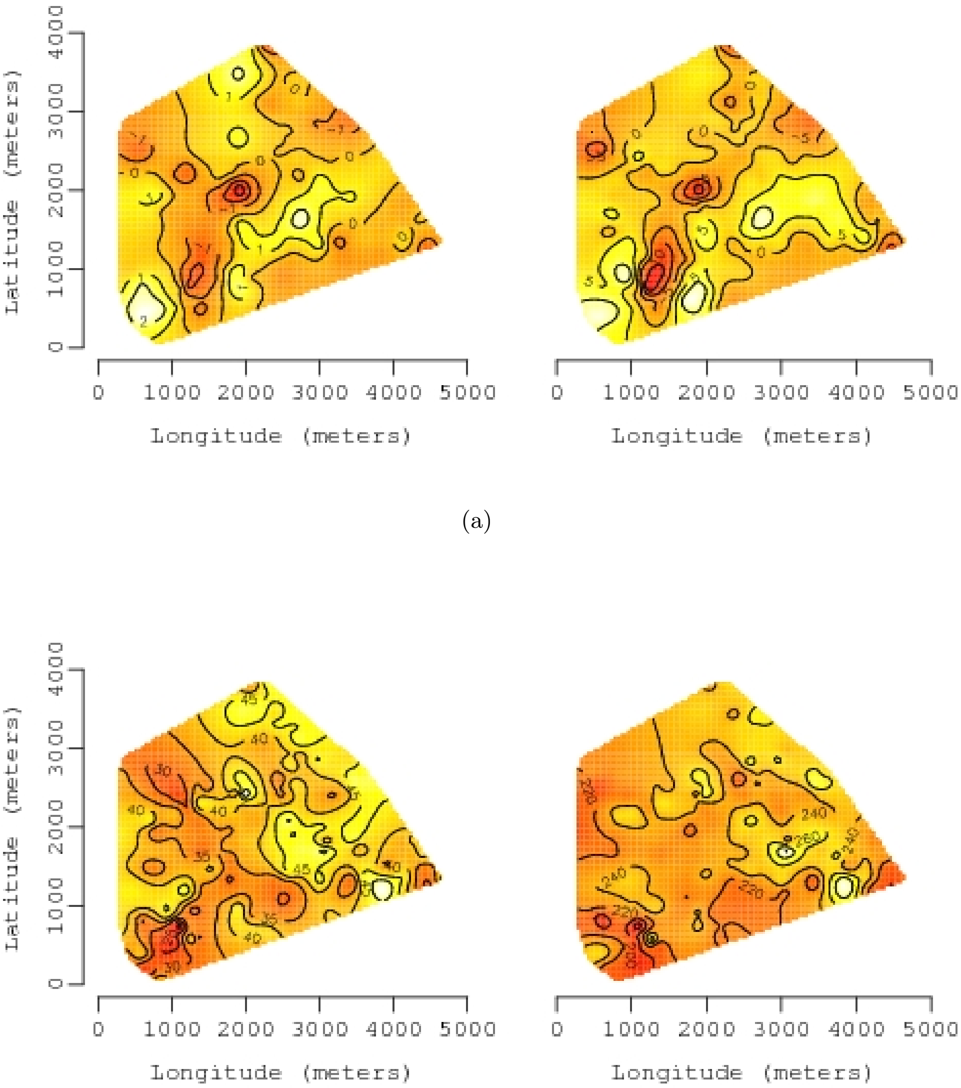
4.7. Forest inventory data analysis results

For parameters estimation, six MCMC chains were run for 10,000 iterations. The six chains allowed for dispersed parameter starting values and because we are interested in the influence of the prior specification on parameter estimates, each chain received a different inverse–Wishart hyperparameter scale matrix. As seen in the synthetic data set analysis, chain mixing occurred within 1,000 iterations. Therefore, 54,000 samples were retained for posterior analysis. Visual interpretation of the changes and resulting parameter estimates suggest that for this data set, the inverse–Wishart scale hyperparamter has negligible influence on chain convergence.

Table 3 provides the credible intervals for each parameter in the model. These intervals show that several regressors help explain the variation in both basal area and biomass per hectare. The significance of the off–diagonal elements $\mathbf{K}_{2,1}$ and $\Psi_{2,1}$ suggests that there is positive spatial and non–spatial association between the conditional response surfaces. Additional

Parameter	Estimates: 50% (2.5%, 97.5%)
$\beta_{1,0}$	-66.912 (-147.658, 14.279)
$\beta_{1,ELEV}$	-0.011 (-0.030, 0.007)
$\beta_{1,TC1}$	1.287 (0.366, 2.184)
$\beta_{1,TC2}$	-1.051 (-1.638, -0.419)
$\beta_{1,TC3}$	1.502 (0.690, 2.293)
$\beta_{2,0}$	-312.771 (-838.557, 207.984)
$\beta_{2,ELEV}$	-0.076 (-0.198, 0.041)
$\beta_{2,TC1}$	6.899 (1.072, 12.701)
$\beta_{2,TC2}$	-4.063 (-7.835, -0.083)
$\beta_{2,TC3}$	6.036 (0.884, 11.141)
$\mathbf{K}_{1,1}$	85.346 (14.806, 152.670)
$\mathbf{K}_{2,1}$	519.223 (20.078, 930.767)
$\mathbf{K}_{2,2}$	3920.109 (316.533, 6863.146)
$\Psi_{1,1}$	71.814 (13.653, 142.901)
$\Psi_{2,1}$	359.778 (11.178, 838.964)
$\Psi_{2,2}$	2686.357 (415.428, 5916.541)
ϕ_1	0.013 (0.003, 0.029)
ϕ_2	0.017 (0.005, 0.029)

Table 3: Percentiles of the posterior distribution of model parameters. β subscripts refer to the response variable and parameter, respectively. Subscripts on $\mathbf{K}$ and Ψ refer to the covariance matrix element. Subscripts on the spatial range parameters, ϕ, refer to the response variable model. Summaries generated from six chains of 9,000 samples.



(a)

(b)

Figure 6: (a) Interpolated surfaces of the predicted random spatial effects for basal area per hectare (left plot) and metric tons of biomass per hectare (right plot), $E[\mathbf{W}^* \mid Data]$. (b) Interpolated surfaces of the posterior predictive distributions for basal area per hectare (left plot) and metric tons of biomass per hectare (right plot), $E[\mathbf{Y}^* \mid Data]$.

insight is gained by converting these off–diagonal covariances to correlations, specifically 0.886 (0.241, 0.952) and 0.84 (0.111, 0.941) are the 50% (2.5%, 97.5%) percentiles of the $\mathbf{K}_{2,1}$ and $\Psi_{2,1}$ elements, respectively. The spatial range estimates in Table 3 do not support a distinction between the responses' spatial dependence structure, therefore, the separable form of this model might be considered.

Given the parameters posterior samples, we can now turn to predicting basal area and biomass for the holdout set of sites marked with triangle symbols in Figure 5(a). Alternatively, we could make predictions for each 30×30 meter pixel in the image stack of derived satellite spectral components and ELEV variable. Again, the predictions are made using the `sp.predict` function and passing in the `ggt.chain.1` object returned from `ggt.sp` along with the new sites' coordinates and covariates.

```
BA.X <- cbind(rep(1, 150), ELEV, TC1, TC2, TC3)
BIO.X <- cbind(rep(1, 150), ELEV, TC1, TC2, TC3)

pred.covars <- mk.mv.X(list(BA.X, BIO.X))

sp.pred <- sp.predict(ggt.chain.1, start=5000, thin=5,
                      pred.coords=coords,
                      pred.covars=covars)
```

The `sp.pred` object contains the posterior predictive samples for $\mathbf{W}^*$ and $\mathbf{Y}^*$. These samples were used to produce the point estimate surfaces in Figure 6. Additionally, we could map some function of the posterior samples. For example, we might map the range between the upper and lower 95% credible interval of $\mathbf{Y}^*$ to understand how uncertainty in prediction varies across the domain.

5. Summary

There is a need within the scientific community for software tools capable of efficiently fitting complex co–varying point level Gaussian spatial process models. The generalized template introduced in Section 3 allows univariate and multivariate Gaussian models to be cast in a common framework. This facilitates model parameter estimation and predictive inference implemented in the **spBayes** package. As detailed in the formal **spBayes** documentation, the `ggt.sp` functions accepts the set of usual covariance functions for modeling spatial dependence, and prior distributions for the variance and covariate parameters.

Although the illustrations presented here consider only bivariate processes, `ggt.sp` and the support functions easily accommodate any number of response variables and associated covariates. Computing time is the only restriction. The computational burden for implementing our template will explode with a large number of locations. This is known as the so–called "big–N" problem in spatial statistics and is an area of active research. Strategies for addressing this problem involve representing the spatial process $\mathbf{W}(\mathbf{s})$ over a smaller set of representative locations (called knots).

We hope future releases of **spBayes** will continue to help fulfill multivariate spatial process modeling needs. In the near-term, we plan to include facilities for modeling big-N problems, spatially varying regressors, multi–resolution or spatially nested models, and non-Gaussian response models.

References

Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J, Du Croz J, Greenbaum A, Hammarling S, McKenney A, Sorensen D (1999). ***LAPACK** Users' Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, third edition. ISBN 0-89871-447-8.

Banerjee S, Carlin BP, Gelfand AE (2004). *Hierarchical Modeling and Analysis for Spatial Data*. Chapman & Hall/CRC, Boca Raton.

Blackford SL, Demmel J, Dongarra J, Duff I, Hammarling S, Henry G, Heroux M, Kaufman L, Lumsdaine A, Petitet A, Pozo R, Remington K, Whaley CR (2002). "An Updated Set of Basic Linear Algebra Subprograms (**BLAS**)." *Transactions on Mathematical Software*, **28**(2), 135–151.

Carlin BP, Louis TA (2000). *Bayes and Empirical Bayes Methods for Data Analysis*. Chapman & Hall/CRC, New York.

Chilés JP, Delfiner P (1999). *Geostatistics: Modelling Spatial Uncertainty*. Wiley, New York.

Christensen OF, Ribeiro, Jr PJ (2002). "**georglm**: A Package for Generalised Linear Spatial Models." *R News*, **2**(2), 26–28.

Cressie N (1993). *Statistics for Spatial Data*. Wiley, New York.

Gelman A (2006). "Prior Distributions for Variance Parameters in Hierarchical Models." *Bayesian Analysis*, **1**(3), 515–533.

Gelman A, Carlin JB, Stern HS, Rubin DB (2004). *Bayesian Data Analysis*. Chapman & Hall/CRC.

Gilks WR, Richardson S, Spiegelhalter DJ (1996). *Markov Chain Monte Carlo in Practice*. Chapman & Hall/CRC, Boca Raton.

Harville DA (1997). *Matrix Algebra from a Statistician's Perspective*. Springer-Verlag, New York.

Majumdar A, Gelfand AE (2006). "Spatial Modeling for Multivariate Environmental Data Using Convolved Covariance Functions." *Technical report*, Institute of Statistics and Decision Sciences, Duke University, Durham, NC.

Møller J (ed.) (2003). *Spatial Statistics and Computational Methods*. Springer-Verlag, New York.

R Development Core Team (2007). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Ribeiro, Jr PJ, Diggle PJ (2001). "**geoR**: A Package for Geostatistical Analysis." *R News*, **1**(2), 14–18.

Schabenberger O, Gotway CA (2004). *Statistical Methods for Spatial Data Analysis*. Texts in Statistical Science Series. Chapman & Hall/CRC, Boca Raton.

Spiegelhalter D, Best N, Carlin B, van der Linde A (2002). "Bayesian Measures of Model Complexity and Fit." *Journal of the Royal Statistical Society B*, **64**(4), 583–639.

Stein ML (1999). *Interpolation of Spatial Data: Some Theory for Kriging*. Springer-Verlag, New York.

Wackernagel H (2003). *Multivariate Geostatistics: An Introduction with Applications*. Springer-Verlag, Berlin.

Acknowledgments

The work of the first author was supported by the NASA Headquarters Earth System Science Fellowship Grant NGT5 and the USDA Forest Service Forest Inventory and Analysis Program. The work of the second and third authors was supported by the National Institutes of Health grant 1R01CA112444-01 and the National Cancer Institute grant 1-R01-CA95955-01, respectively. The authors are grateful to Galin Jones and two anonymous reviewers for their many helpful comments and suggestions.

Affiliation:

Andrew O. Finley
Department of Forest Resources
University of Minnesota
1530 Cleveland Ave. N.
St. Paul, MN 55108, United States of America
E-mail: afinley@stat.umn.edu
URL: http://blue.fr.umn.edu/

Sudipto Banerjee
Division of Biostatistics
School of Public Health
University of Minnesota
E-mail: sudiptob@biostat.umn.edu

Bradley P. Carlin
Division of Biostatistics
School of Public Health
University of Minnesota
E-mail: brad@biostat.umn.edu

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/
Volume 19, Issue 4 *Submitted:* 2006-10-29
April 2007 *Accepted:* 2007-04-24