A Review on Non-stationary Modeling With Sparsity for Spatial Data via the Basis Graphical Lasso

Course Project for MTH643A: Spatial Statistics

Submitted by: Aritra Basak (221282), Paulomi Das (221366), Pratyusha Bala (221372), Sambit Das (221406)

Supervisor: Dr. Arnab Hazra

Department of Mathematics and Statistics
Indian Institute of Technology Kanpur

Date of Submission:
November 14, 2023
Academic Year 2023–24

Acknowledgement

We would like to extend our sincere gratitude to Dr. Arnab Hazra, our esteemed project guide, for his invaluable guidance, support, and consistent efforts throughout the duration of this project. His expertise, encouragement, and dedication were instrumental in helping us navigate the challenges we encountered.

His commitment to our project went beyond the classroom, as he generously shared his knowledge and insights, contributing significantly to our understanding of the subject matter. We are truly thankful for his mentorship, which has been a driving force behind the successful completion of our project.

– (Aritra, Sambit, Pratyusha, Paulomi)

Contents

1 Introduction

The use of basis expansion with random coefficients to express the stochastic variation component in modern spatial models is a powerful and flexible approach for capturing and modeling complex spatial processes. In this paper the authors have introduced a novel approach that allows for non-stationarity in any model with this framework, and is easily adaptable to large datasets. The method allows for straightforward graphical interpretations of the conditional independence structure of the stochastic coefficients.A general spatial statistical model for an observational process Y(s) with $s \in R^d$ can be written as,

$$Y(s) = \mu(s) + Z(s) + \epsilon(s)...(1)$$

which decomposes the observations into a mean function μ, a spatially correlated random deviation Z, and a white noise process ϵ. Here, we assume, $Z(s) = \sum_{j=1}^{l} c_j \phi_j(s)...(2)$ for fixed basis functions $\phi_1, ..., \phi_l$ and stochastic coefficients $c = (c_1, ..., c_l)^T \sim N(0, Q^{-1})$.

1.1 Distinction from other methods

This model setup is distinct from nonspatial methods like the graphical lasso which require direct realizations of c to estimate the structure of the undirected graphical model Q ,and lasso regression which estimates Q by adding an $l1$penalty to the negative loglikelihood to encourage sparsity. In this problem, extending the $l1$ penalization framework to the basis graphical lasso has given rise to a non-convex optimization problem to fit a graphical model to c, which the authors have solved through an MM algorithm by iteratively solving the graphical lasso. Hence, those previous methods are not applicable in this case.

2 Methods

For ease of exposition we assume, $\mu(s) = 0$. Then the model becomes, .365“ “ For m independent realizations $Y_1(s), ..., Y_m(s)$ at spatial locations $s = s_1, ..., s_n$ let $Y_i = (Y_i(s_1), ..., Y_i(s_n))^T$. Thus, matrix representation of the model is

$$Y_i = \Phi c_i + \epsilon_i, i = 1, ..., m...(3)$$

where Φ is an $n \times l$ matrix with $(i,j)th$ entry $= \phi_j(s_i), c_i = (c_{i1}, ..., c_{il})^T$ are m independent, mean zero-variate Gaussian random vectors with precision matrix Q ,and $\epsilon_i = (\epsilon_i(s_1), ..., \epsilon_i(s_n))^T$ are m independent realizations of the white noise process with mean zero and nugget $\tau^2 > 0$.

This model is nonstationary and computations can be sped up by particularly choosing compactly supported basis Φ and a sparse Q.

2.1 Basis Graphical Lasso

Our aim is to estimate the precision matrix Q assuming that it follows a graphical structure. The graph structure is represented by an undirected graph where nodes correspond to the random variables, and edges indicate dependencies between them. If there is an edge between nodes i and j, it means the covariance between i and j is non-zero, implying that variables i and j are dependent given all other variables in the model.

We introduce sparsity to the model to support the fact that the response at any location is related to the response at it's neighboring locations. Here we use the $l1$ (lasso) penalty to capture the most significant or the relevant dependencies while pushing the less relevant ones to zero.

The precision matrix Q is estimated by minimizing the negative log likelihood of the data with an added $l1$ penalty term to induce sparsity. Say we have m realizations of the response $Y_1, Y_2, ..., Y_m$. The negative log-likelihood equation can be written ignoring the multiplicative and additive constants as,

$$\log\det(\Phi Q^{-1}\Phi^T + \tau^2 I_n) + tr(S(\Phi Q^{-1}\Phi^T + \tau^2 I_n)^{-1}), ...(4)$$

, where $S = (1/m)\sum_{i=1}^{m} Y_i Y_i^T$ is an empirical covariance matrix. $Q \geq 0$ indicates that it must be a positive semi-definite matrix and $||\lambda \cdot Q||_1 = \sum_{i,j} \Lambda_{ij}|Q_{ij}|$ is a penalty term that enforces sparsity on the elements of Q. Higher the values in the matrix lambda, the more sparse the estimated precision matrix will be. The diagonal elements of lambda are 0 since sparsity is encouraged only in the off-diagonal elements of Q.

The objective function is difficult to optimize because of it's dependence on the spatial dimension n and the nested inverses surrounding Q. The following proposition will help overcome this difficulty.

Proposition: The minimizer of (4) is also the minimizer of

$$\log\det(Q+\tau^{-2}\Phi^T\Phi) - \log\det Q \ - tr(\tau^{-4}\Phi^T S\Phi(Q+\tau^{-2}\Phi^T\Phi)^{-1}) + ||\Lambda \cdot Q||_1 ...(5)$$

2.2 Optimization Approach

The optimization approach for your problem involves dealing with a function that has components that are either concave or convex. Here's a explanation of how to address this:

In the case where $l = 1$,equation (5) represents a univariate function that is twice differentiable on the positive real line. It's possible to choose appropriate values for $\Phi^T\Phi$, $\Phi^T S\Phi$, and τ^2 so that the second derivative of this function has a negative value at some point along the positive real line. This characteristic makes equation (5) nonconvex on the interval $Q \geq 0$.

We can, however, show that the four summands in (5) are concave, convex, concave, and convex, respectively, on ($Q \geq 0$); see Appendix A for details. Therefore, the objective function in (5) can be written as

$$\arg\min_{Q \geq 0} (f(Q) + g(Q) + \|\Lambda \cdot Q\|_1) \qquad (6)$$

where, $f(Q) + ||\Lambda \cdot Q||_1$ is convex, and $g(Q)$ is concave and differentiable.

A natural approach for this nonconvex problem is a **Difference-of-convex (DC) program**(Dinh Tao and Le Thi 1997) where we iteratively linearize the concave part $g(Q)$ at the previous guess Q_j and solve the resulting convex problem:

$$\arg\min_{Q\geq 0} \left(f(Q) + \mathrm{tr}(\nabla g(Q_j)Q) + \|\Lambda \cdot Q\|_1\right) \tag{7}$$

In optimizing the DC framework, the **MM algorithm** (Hunter and Lange, 2004) proves useful.
It relies on majorization, where a function $h(\theta)$ is dominated by $m(\theta|\theta^*)$ at θ^* if $h(\theta) \leq m(\theta|\theta^*)$ for all θ, and $h(\theta^*) = m(\theta^*|\theta^*)$. Instead of directly minimizing the complex $h(\theta)$, the MM algorithm addresses a series of minimization problems, focusing on minimizing the majorizing function at each step.

$$\theta_{j+1} = \arg\min_{Q\geq 0} m(\theta|\theta_j) \tag{8}$$

Combining (11) with the definition of a majorant yields the inequality

$$h(\theta_{j+1}) \leq m(\theta_{j+1}|\theta_j) \leq m(\theta_j|\theta_j) = h(\theta_j)$$

and thus, the algorithm is compelled toward a local minimum (or saddle point) of $h(\theta)$. DC programming, also known as the concave-convex procedure, belongs to the subclass of MM, where the supporting hyperplane inequality $g(\theta) \leq g(\theta_j) + < \nabla g(\theta_j)(\theta - \theta_j) >$ is utilized to construct a majorizing function when $h(\theta)$ is expressed as the sum of a concave differentiable function $g(\theta)$ and a convex function. In other words, when $h(\theta)$ is a difference of convex functions. An added advantage within the DC framework is that the majorizing function is convex by construction, and thus, we solve a series of convex optimization problems in each step of (8).

In our likelihood function, the convex part is

$$f(Q) = -\log\det Q$$

and the concave part is

$$g(Q) = \log\det(Q + \tau^{-2}\Phi^T\Phi) - \mathrm{tr}(\tau^{-4}\Phi^T S\Phi(Q + \tau^{-2}\Phi^T\Phi)^{-1})$$

So the DC algorithm (7) becomes

$$Q_{j+1} = \arg\min_{Q\geq 0}(-\log\det Q + \mathrm{tr}\,(\nabla g(Q_j)Q) + \|\Lambda \cdot Q\|_1)), \tag{9}$$

The inner minimization problem in (9) is well-studied and known in statistics as the **graphical lasso** problem.

Traditionally, the graphical lasso is used to estimate an undirected graphical model of a multivariate Gaussian vector $\mathbf{c}$ under the assumption that we observe $\mathbf{c}$ directly, without noise. The standard graphical lasso estimate is obtained from the penalized negative log-likelihood

$$\arg\min_{Q\geq 0} -\log\det Q + \mathrm{tr}(S_c Q) + \|\Lambda \cdot Q\|_1, \tag{10}$$

where S_c is the sample covariance of $\mathbf{c}$. In summary, we have shown that the BGL (4) for estimating the graphical structure of Q given realizations from $\Phi_{\mathbf{c}} + \epsilon$ can be discerned through a concave-convex procedure where the inner solve is the graphical lasso (10) with a "sample covariance" matrix depending upon the previous guess Q_j.

QUIC Algorithm (Hsieh et al. 2014) advances in solving (10) in recent years have stemmed from the use of second-order methods that incorporate Hessian information instead of simply the gradient.

Briefly, the QUIC algorithm uses coordinate descent to search for a Newton direction based on a quadratic expansion about the previous guess and then an Armijo rule to select the corresponding stepsize. During the coordinate descent update, only a set of free variables are updated, making the procedure particularly effective when Q is sparse.

2.2.1 Estimating the Nugget Variance

In practice, the algorithm (9) requires a fixed estimate $\hat{\tau}^2$. To achieve this, we revisit (4), now expressed as $f(Q, \tau^2) = \log\det(Q + \tau^{-2}\Phi^T\Phi) - \log\det Q - tr(\tau^{-4}\Phi^T S\Phi(Q + \tau^{-2}\Phi^T\Phi)^{-1} + n\log\tau^2 + \tau^{-2}tr(S)$...(14)

We jointly minimize f over τ^2 and α, assuming $Q = \alpha I_l$ with $\alpha > 0$. The estimates $\hat{\tau}^2$ and $\hat{\alpha}$ are obtained through an **L-BFGS optimization** routine using the 'optim' function in R. Empirically, in the simulation study below, this approach works very well. However, jointly estimating a full model of Q and τ^2 is a complicated task.

2.2.2 Estimating the Penalty Weight

The remaining specification involves determining the penalty weight matrix Λ. One approach, inspired by Bien and Tibshirani (2011), employs a likelihood-based cross-validation method to select Λ when estimating a sparse covariance matrix. Formally, we employ k folds and consider t penalty matrices $(\Lambda_1, \ldots, \Lambda_t)$. The estimate $\hat{Q}_\Lambda(S)$ is derived by applying our algorithm to the empirical covariance $S = \frac{1}{m}\sum_{i=1}^m Y_i Y_i^T$ with the chosen penalty Λ. We seek Λ so that $\alpha(\Lambda) = l(\hat{Q}_\Lambda(S), S)$ is minimized, where

$$l(Q, S) = \log\det(Q+\tau^{-2}\Phi^T\Phi) - \log\det Q - \mathrm{tr}(\tau^{-4}\Phi^T S\Phi(Q+\tau^{-2}\Phi^T\Phi)^{-1} \quad (15)$$

represents the unpenalized version of (7).The cross-validation approach involves partitioning $\{1, \ldots, m\}$ into disjoint sets $\{\Lambda_1, \ldots, \Lambda_k\}$ and selecting

$$\hat{\Lambda} = \arg \min_{\Lambda \in \{\Lambda_1, \ldots, \Lambda_t\}} \hat{\alpha}(\Lambda)$$

, where

$$\hat{\alpha}(\Lambda) = \frac{1}{k}\sum_{i=1}^k l(\hat{Q}_{\hat{\Lambda}}(S_{A_i^c}), S_{A_i}).$$

2.2.3 Initial Guess and Convergence

The non convex nature of this problem makes common convergence criteria for convex optimization unsuitable. Instead, we define convergence when

$$\frac{\|\mathbf{Q}_{j+1} - \mathbf{Q}_j\|_F}{\|\mathbf{Q}_j\|_F} < \epsilon$$

for some loose tolerance $\epsilon = 0.01$. We initiate the algorithm with $\mathbf{Q}_0 = \mathbf{I}$, which can lead to large diagonal entries since the diagonal penalty weights of Λ are set to zero.

3 Simulated Data studies

In this section, three different perspectives of the proposed model is analysed. The first is a comparison study against two possible alternative estimation procedures, the second a timing study to illustrate the trade-off between realizations and graph dimension,and the third a simulation study under different basis and graph structures.

3.1 Comparison Study

Let us consider two alternative methods to the BGL for estimating the precision (or covariance) matrix of the Gaussian vector c under the basis model.

A naive but straight forward approach involves projecting the data onto the basis and using the projected data with standard shrinkage methods for estimating sparse precision matrices. Following notation in (4),The estimated regression coefficient vector, denoted as $\hat{c}$, is calculated through the method of least squares as $(\Phi^T\Phi)^{-1}\Phi^T Y_i$, where $(\Phi^T\Phi)^{-1}\Phi^T$ is the projection matrix and Y_i is the response vector. If Φ is not full rank, we cannot consider this least squares projection - a ridge regression maybe more suitable, for example. Once $\hat{c}$ is created, its sample covariance S is substituted into the graphical lasso; we call this approach the **regression method**. For small samples, this approach is expected to be statistically inefficient as it does not explicitly use any likelihood information and moreover fails to incorporate the white noise process ϵ.

Another possible approach in the style of **fixed rank kriging** is to retrieve $K = Q^{-1}$ by minimizing the loss function

$$argmin_{K \geq 0}||\Phi K \Phi^T + \tau^2 I_n - S||_F (16)$$

as proposed by Cressie and Johannesson (2008). From equation (3.8) in Cressie and Johannesson (2008), the optimal parameter estimate is $\hat{K} = R^{-1}Q^T(S - \tau^2 I_n)Q(R^{-1})^T$ where $\Phi = QR$ is the QR decomposition of Φ. Again,if Φ is not full rank, we cannot consider this method. The R package FRK is designed for one realization (So, FRK is used only, when m = 1), and (16) is an analogous estimate for multiple realizations. Here, we have used the “unstructured” parameterization for the basis-function covariance matrix K, for more comparable nonparametric estimate. The connection to graphical modeling here is lost,as we cannot expect a sparse inverse of the estimated covariance matrix.

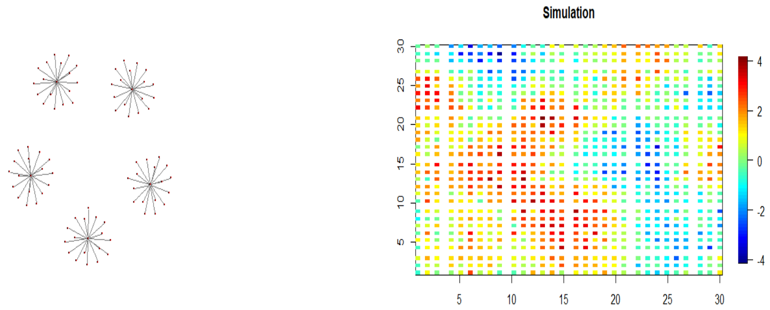

Figure 1: Graphical Structure of the precision matrix Q and a realization from the basis model

Table 1: Comparing methods: loss is the fixed rank kriging approach for multiple realizations (FRK is used when m=1), regression is the projection estimate, and BGL is the proposed basis graphical lasso.

Statistics	m	Loss/FRK	Regression	BGL
Frobenius error	1	8267	2.38	12.76
	5	156.04	1.11	1.32
	10	98.52	0.71	0.80
	20	49.84	0.47	0.78
	50	21.69	0.51	0.79
KL divergence	1	1,96,072	104.38	**78.2**
	5	-	45.50	**44.0**
	10	-	37.19	**27.8**
	20	-	33.08	**17.2**
	50	-	36.54	**12.4**

Table 2: NOTE:Values are means based on 5 independent trials. Missing values for Loss/FRK are due to numerically singular covariance matrix estimates.

3.1.1 The setup of the experiment is as follows:

We have considered, n = 900 observations on the two-dimensional grid $(i, j)_{i,j=1}^{30}$. We have used a total of $l = 90$ bisquare basis functions taken from the FRK package using two resolutions. We consider the graph Q to be a hub graph where the nodes are separated into groups and each member of that group is only neighbors with a central node. The hub graph is generated with default parameters from the R package **huge** (Zhao et al. 2015) for high-dimensional undirected graph estimation. The graphical structure is inherently tied to the spatial registration of the basis functions, but we are ignoring this and simply focusing on the ability of the methods to recover the precision matrix.

We have fixed the noise-to-signal ratio $\tau^2/(tr(\Phi Q^{-1}\Phi^T)/n)$ at 0.1 to determine the true nugget variance τ^2. See Figure 1. for an illustration of the graph structure and a realization of the process Φc. The penalty matrix is populated with 0.25 (i.e. lambda = 0.25); we found that regularizing the diagonal helped when the number of realizations was small.

To compare these estimators, we conduct 5 trials and report summary statistics based on the Frobenius norm $||\hat{Q} - Q||_F/||Q||_F$ and the Kullback–Leibler (KL) divergence $tr(\hat{Q}Q) - log|\hat{Q}Q| - l$. Results are shown in Table 1.

The estimates based on minimizing the loss function (16) were poorly behaved in comparison to the BGL method, and the versions which did not directly call the FRK package produced singular matrices. Whereas, the BGL method is more or less comparable to the regression-based idea.

3.2 Timing Study

In examining the timing results for solving the BGL equation (6) through our DC algorithm (10), the runtime is primarily affected by several factors. These include the number of basis functions (denoted as Φ), the quantity of realizations (m, indicative of the sample covariance S's quality), the sparsity of Q, and the chosen penalty. It's important to note that the spatial dimension (n) no longer plays a role in estimating Q once $\Phi^T\Phi$ and $\Phi^T S\Phi$ are stored. The latter matrix, $\Phi^T S\Phi$, is crucial for directly incorporating data into the BGL and should be computed using equation (8) to avoid storing the complete sample covariance matrix S. The QUIC algorithm ensures quadratic convergence for each inner solve of (10) (Hsieh et al. 2014). However, determining the overall complexity of (6) remains challenging.

The spatial domain in consideration is the unit square $[0, 1] \times [0, 1]$, and basis functions are compactly supported Wendland bases from the LatticeKrig R package. We observe data at $n = 2500$ uniformly randomly sampled spatial locations with a noise-to-signal ratio of 0.1. The fixed penalty matrix is populated with 0.2 and zeros on the diagonal - no cross-validation is performed. The initial guess is the identity matrix $Q_0 = I_l$, as in the rest of the article. We stop the BGL at a relative error of 0.01.

Figure 2 illustrates the time taken by the BGL. Broadly speaking, the computational time tends to escalate more rapidly than linearly concerning the number of basis functions. Here, we see that having multiple realizations helps reduce computation time due to a more stable estimate of the sample spatial covariance matrix S.

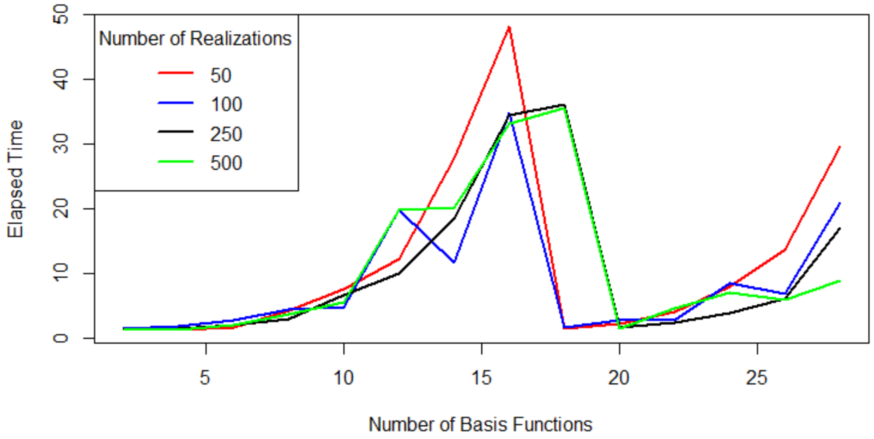

Figure 2: Illustrating the time to convergence for the BGL algorithm.

3.3 Simulation Study

This section concludes with a series of simulation studies designed to evaluate the effectiveness of our proposed algorithm in recovering unknown precision structures under the model (4). We used localized basis functions with spatially compact support over the entire domain. Within each class, we explore various types of precision structures commonly encountered in graphical modeling literature.

For each combination of n,Φ and Q, the noise-to-signal ratio is consistently set at 0.1, thereby determining the true nugget variance τ^2 . The estimated nugget variance, denoted as $\hat{\tau}^2$, is obtained from Section 2.2.1. Despite the population mean being assumed to be zero in these simulations, it is typically unknown in practical scenarios. Therefore, we consistently employ the standard unbiased estimator S, which involves an empirical demeaning process. This approach is chosen to better reflect practical implications compared to using a known mean.

3.3.1 Local Basis

To start, we examine a localized problem where we utilize a basis comprising compactly supported functions arranged on a grid using the LatticeKrig model setup (Nychka et al., 2015). This setup involves compactly supported Wendland functions, with the range of support configured so that each function overlaps with 2.5 other basis functions along axial directions. The model basis functions are designed for either a single level or a multiresolution model. In the single level setup, functions are positioned on a regular grid. In the multiresolution setup, higher levels of resolution are achieved by increasing the number of basis functions and nodal points (e.g., the second level doubles the number of nodes in each axial direction). The precision matrix Q is configured to follow a stationary spatial autoregressive structure; for additional information, refer to Nychka et al. (2015).

We set the variance of the multiresolution levels to exhibit behavior resembling an exponential covariance by choosing the parameter ν to be 0.5. In the LatticeKrig framework, the precision matrix Q is constructed based on a spatial autoregression parameterized by the value α, which is fixed at $\alpha = 4.05$. For simplicity, when constructing the Wendland bases, we opt not to use a buffer region; in other words, there are no basis functions centered outside of the spatial domain. The R package LatticeKrig is utilized to establish the aforementioned basis and precision matrices. A total of $m = 500$ realizations from the process (4) under this model are generated, and we replicate this entire spatial data-generation process across 30 independent trials.

The spatial domain is defined as $[0, 1] \times [0, 1]$, and n observation locations are randomly chosen uniformly within this domain for various sample sizes $n \in \{100^2, 150^2, 200^2\}$. In the case of the single-level Wendland basis, we employ $l \in \{100, 225, 400\}$ basis functions. To replicate these dimensions in the multiresolution basis, we use $l \in \{119, 234, 404\}$, which, respectively, corresponds to (1) four multiresolution levels, with the coarsest level containing two Wendland basis functions, (2) three multiresolution levels, with the coarsest level containing four Wendland basis functions, and (3) four multiresolution levels, with the coarsest level containing three Wendland basis functions.

We parameterize the penalty matrix Λ according to

$$\Lambda_{ij} = \begin{cases} \lambda & i \neq j \\ 0 & i = j \end{cases} \tag{1}$$

allowing for free estimates of the marginal precision parameters. We employ a 5-fold cross-validation, as detailed in Section 2.2.2, to select a penalty matrix Λ from eight equally spaced values ranging from 0.005 to 0.1. The optimal value is then utilized with the complete set of simulated realizations to generate the best estimate $\hat{Q}$. To assess the validity of our proposed estimation approach, we present several summary statistics, each averaged over the 30 trials. These statistics include: the Frobenius norm $||\hat{Q} - Q||_F / ||Q||_F$, the KL divergence $tr(\hat{Q}Q) - log|\hat{Q}Q| - l$, the percentage of zeros in Q that were overlooked by $\hat{Q}$, the percentage of nonzero elements in Q that were overlooked by $\hat{Q}$, the difference between the estimated nugget effect τ^2 and the true nugget effect τ^2, and the ratio of the estimated and true negative log-likelihoods $f(\hat{Q}, \hat{\tau}^2)$ and $f(Q, \hat{\tau}^2)$, where f is defined in Section 2.2.1.

Table 3: Simulation study results for the single level case.

n	l	Frob	KL	%MZ	%MNZ	$\widehat{\tau^2} - \tau^2$	$f(\hat{Q}, \hat{\tau}^2)/f(Q, \tau^2)$
10,000	100	0.41	6.5	21	5	0.000008	1.0001
	225	0.49	26	20	2	-0.000042	1.00081
	400	0.72	110	4.7	0.8	0.00016	0.998709
22,500	100	0.38	5.6	22	6	0.0000093	1.00005
	225	0.43	21	21	2	-0.0000041	1.00037
	400	0.65	82	4.8	1	-0.000033	0.999366
40,000	100	0.37	5.3	23	6	0.0000084	1.00002
	225	0.41	19	22	3	0.0000038	1.00021
	400	0.61	70	4.9	1	-0.0000079	0.999617

Table 4: NOTE: Scores are averaged over 30 independent trials. Each column represents the number of observation samples, number of basis functions, Frobenius norm, KL divergence, percent of true zeros missed, percent of true nonzeros missed, estimated nugget minus true nugget, and the estimated negative log-likelihood divided by the true negative log-likelihood.

3.3.2 Comments

Tables 3 and 5 contain results from this simulation study. We see that estimating the nugget effect τ^2 by treating the process as stationary is quite accurate; here and in the rest of the article, the unreported standard deviations of the averaged differences $\widehat{\tau^2} - \tau^2$ are many orders of magnitude smaller than the magnitude of the true nugget effect.

Estimates under the multi resolution basis are clearly lackluster when compared to the single resolution counterpart. The Frobenius norm and KL divergence tend to increase with the size of l, but this is to be expected as the dimensions of the target precision matrix Q grow in l. The percentage of zeros in Q that are missed (i.e., nonzero) in $\hat{Q}$ drops sharply as l increases to 400, but this is the consequence of a harsher penalty weight matrix selected in the cross-validation scheme.

4 Application

The Topography Weather data set (Oyler et al. 2015) contains observed air temperatures from a set of observation networks over the continental United States. They have considered daily minimum temperatures during the month of June from 2010 to 2014, giving a total of m = 150 realizations. There are no missing values, and n = 4577 spatial locations. Figure 3 shows an example day of data on June 1, 2010.

They have worked with minimum temperature residuals after removing a pixelwise mean over realizations and also transformed the raw spatial coordinates with a sinusoidal projection. This model uses Wendland basis functions

Table 5: Simulation study results for the multiple level case.

n	l	Frob	KL	%MZ	%MNZ	$\hat{\tau^2} - \tau^2$	$f(\hat{Q}, \hat{\tau}^2)/f(Q, \tau^2)$
10,000	119	0.82	645	5.6	7	-0.000035	1.00002
	234	0.89	2020	6.4	4	-0.00013	1.00022
	404	0.85	3790	0.07	3	-0.0002	0.999782
22,500	119	0.77	640	7.1	6	-0.0000078	1
	234	0.85	2050	10	3	-0.000043	1.00017
	404	0.93	4280	0.085	2	-0.0001	0.99983
40,000	119	0.79	635	7.8	6	0.0000012	0.999998
	234	0.84	1940	11	3	-0.000015	1.00011
	404	0.94	4440	0.087	2	-0.000046	0.99986

Table 6: NOTE: Scores are averaged over 30 independent trials. Each column represents the number of observation samples, number of basis functions, Frobenius norm, KL divergence, percent of true zeros missed, percent of true nonzeros missed, estimated nugget minus true nugget, and the estimated negative log-likelihood divided by the true negative log-likelihood.

centered at nodes displayed in Figure 3. opt for $l = 1160$ functions using a single level of resolution. The nodal grid and Wendland functions are chosen to match up with a LatticeKrig model specification but with relaxed assumptions on the precision matrix governing the random coefficients. The nugget estimate is $\hat{\tau^2} = 2.18$. The penalty matrix Λ is parameterized according to $\Lambda = \lambda D$, where D is the distance matrix of node points and λ is selected to be 7 from $\lambda \in \{1, 2, ..., 30\}$ using the likelihood-based cross-validation scheme in Section 2.2.2.

Figure 4 and **5** show graphical model neighborhoods and estimated correlation functions centered at locations in Utah and Kansas. Clearly, anisotropy and nonstationarity is present in the estimated correlation functions with greater north-south directionality of correlation, while the neighborhood structure for the Utah nodal point (a) displays greater complexity than the relatively nearby neighbors of the nodal point in the midwest Kansas (b). Figure 4 suggests that the stationary spatial autoregressive assumption whis is underlying both LatticeKrig and the standard SPDE approach are inappropriate for these data. Due to lack of data availability over the ocean there is an identifiability problem with this method. Since several of the Wendland basis functions lie over the ocean where there is no observed data, we cannot expect the algorithm to give reasonable estimates for the diagonal elements of Q corresponding to those nodes. Moreover, the diagonal of the penalty matrix Λ is identically zero, and thus the corresponding diagonal elements of Q remained unchanged no matter the initial guess Q_0, which is fixed at $Q_0 = I_1$.

Comparison: Here the authors have compared BGL model against the analogous LatticeKrig model using the same nodal grid and same Wendland basis functions but with the spatial autoregressive precision matrix of LatticeKrig.

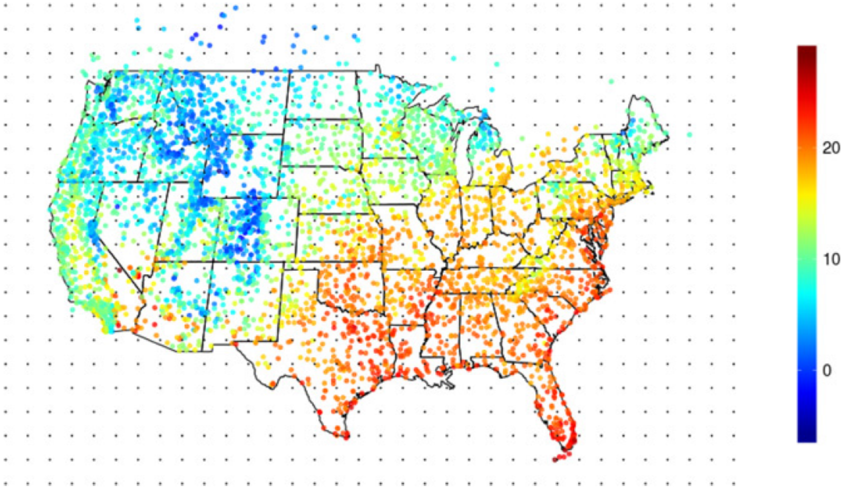

Figure 3: Minimum temperature (Celsius) on June 1, 2010, overlaid with a grid of basis function nodes.

In particular, the central a.wght parameter is estimated at 4.495, and the nugget variance $\tau^2 = 2.23$ is close to BGL estimate. The smoothness parameter nu in the LatticeKrig setup is set at 0.5, which is a typical assumption for observational temperature data.

However, using this few basis functions downplays the capabilities of LatticeKrig, so the authors have included a multiresolution LatticeKrig model created with 3 levels. The coarsest of this matches the nodepoints from the smaller example. In total, there are 15,394 basis functions over the 3 levels, and the estimated a.wght and nugget variance parameters are 4.161 and 2.05, respectively.

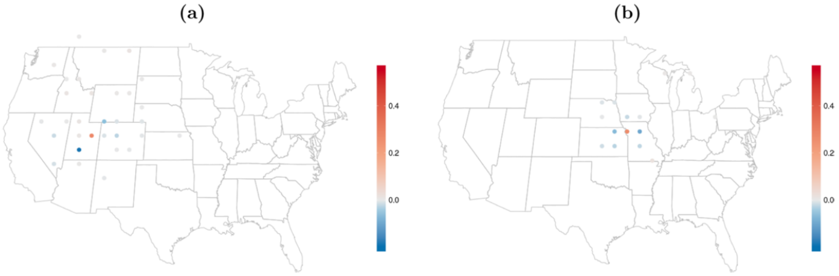

Figure 4: Displaying the neighborhood structure of Q, where the two center points are located in central Utah (a) and near Kansas City (b). Neighbors are colored according to the corresponding nonzero elements in Q.

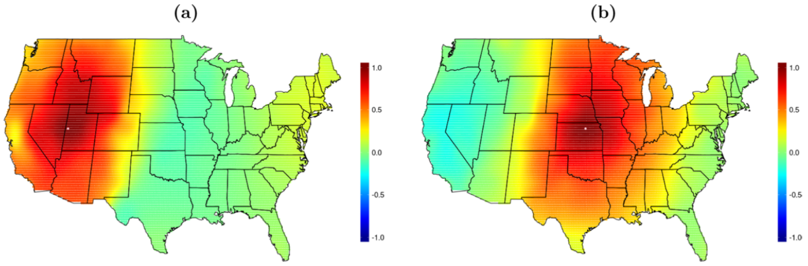

Figure 5: . Estimated spatial correlation functions centered at the pink-colored locations in Utah (a) and Kansas (b).

The three models are compared based on cross-validation prediction accuracy and standard Akaike information criterion. Randomly 400 locations are hold out and the kriging predictive distribution at these locations for each day of data is calculated . Point predictions are compared using the average RMSE. To compare the predictive distributions, two proper scoring rules are applied: the continuous ranked probability score (CRPS), which quantifies the quality of the marginal predictive distributions at each location separately, and the (negative) log score (Gneiting and Raftery 2007) which is a measure of the quality of the joint predictive distribution over all validation locations simultaneously. To calculate the AIC, we note that the number of degrees of freedom of a spatial model can be identified with the trace of the spatial smoothing hat matrix (Nychka 2000).

Table 7 below contains the averaged scores over all days. Both LatticeKrig models give a marginally better CRPS despite being designed for a single spatial field m = 1, but these scores only measure the marginal behavior of the predictive distributions. The AIC and log scores quantify the quality of the joint distributions and suggest that the BGL more accurately represents such joint distributions of the process than the LatticeKrig models.

Table 7: Cross-validation results comparing the proposed basis graphical lasso (BGL) to two versions of LatticeKrig on the TopoWX data

	RMSE	CRPS	Negative log score	AIC
BGL	1.47	2.15	714.4	9018.5
Single-level Latticekrig	1.48	2.13	1084.6	9063.2
Multi-level LatticeKrig	1.46	2.14	997.6	9341.6

5 Conclusion

In this study, the authors have presented a new method for estimating the precision matrix of the random coefficients in a basis representation model commonly used in spatial statistical research. Only assumption about the precision matrix is its sparsity. When the basis functions are aligned to a grid, the precision entries can be viewed as a spatial Gaussian Markov random field. Additionally, interpretations in terms of graphical models remain applicable when using global bases.

The proposed BGL estimator aims to minimize a negative log-likelihood equation with an $l1$ penalty. It is demonstrated that the optimization problem can be viewed as involving a sum of convex and concave functions. This observation suggests a DC algorithm, wherein we iteratively linearize the concave part at the previous estimate and solve the resulting convex problem. In our case, the linearization leads to a graphical lasso problem, with the ”sample covariance” depending on the previous estimate. Graphical lasso problems are well-studied, and several user-friendly R packages, such as the second-order method QUIC, are available. This method holds significant practical value in spatial data analysis, providing a nonparametric, penalized maximum likelihood estimate of Q. This estimate can then be utilized in kriging or simulation, with computational complexity $O(nl^2)$ under the basis model.

In the illustrative data instances, we note that the proposed method demonstrates competitive performance with established alternatives such as LatticeKrig when it comes to marginal predictions. Notably, it brings about substantial improvements in the accuracy of joint predictive distributions. Furthermore, the model yields interpretable fields, facilitating the examination of graphical neighborhood structures or implied nonstationary covariance functions. For future investigations, avenues may include exploring different penalty approaches, increasing the number of basis functions to accommodate various levels of resolution more effectively, or extending these concepts into the realm of space-time modeling.

6 Reference

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, DOI: 10.1080/10618600.2014.914946.

Heish, C.-J., Sustik,M. A., Dhillon, I. S., and Ravikumar, P. (2014), “QUIC: Quadratic Approximation for Sparse Inverse Covariance Estimation,” Journal of Machine Learning Research, 15, 2911–2947. [378,379]

Rue, H., and Held, L. (2005), Gaussian Markov Random Fields: Theory and Applications, Monographs on Statistics and Applied Probability, Boca Raton, FL: Chapman Hall/CRC.