

Bayesian Transfer Learning for Artificially Intelligent Geospatial Systems: A Predictive Stacking Approach

Luca Presicce

*Department of Economics, Management and Statistics
University of Milano-Bicocca
Milan, Italy*

L.PRESICCE@CAMPUS.UNIMIB.IT

Sudipto Banerjee

*Department of Biostatistics
University of California, Los Angeles
Los Angeles, CA 90025, USA*

SUDIPTO@UCLA.EDU

Editor: Christoph Lampert

Abstract

Building artificially intelligent geospatial systems requires rapid delivery of spatial data analysis on massive scales with minimal human intervention. Depending on their intended use, learning about underlying spatial processes can also involve model assessment and uncertainty quantification. We devise transfer learning frameworks for deployment in artificially intelligent systems, where a massive data set is split into smaller data sets that stream into the analytical framework to propagate learning and assimilate learning for the entire data set. Specifically, we develop Bayesian predictive stacking for multivariate spatial data and demonstrate rapid automated probabilistic learning from massive spatial data sets. We illustrate the effectiveness of our approach through extensive simulation experiments and through the analysis of a massive dataset on vegetation index that are indistinguishable from traditional (and more expensive) statistical approaches.

Keywords: Bayesian predictive stacking, climate science, Gaussian process, geospatial systems, matrix-variate spatial models, transfer learning

1. Introduction

Geospatial artificial intelligence (GEOAI) is a rapidly evolving discipline at the interface of machine learning and spatial data science that attempts to harness the analytical capabilities of Artificial Intelligence (AI) to analyze massive amounts of geographic data for data-driven scientific discoveries in the environmental and physical sciences. The area, while still fledgling, presents opportunities to devise essential data analytic tools that should comprise an artificially intelligent “geospatial” system. This manuscript focuses primarily on statistical learning tools for artificially intelligent GEOAI systems.

What should be the role of probabilistic learning in GEOAI? Formal inference for spatial random fields enjoys a venerable intellectual presence spanning several decades of theoretical developments within classical and Bayesian paradigms (see, e.g., Cressie, 1993; Stein, 1999; Gelfand et al., 2010; Cressie and Wikle, 2011; Gelfand and Banerjee, 2017; Banerjee et al., 2025b). Statisticians have built richly structured hierarchical models in pursuit of “full”

probabilistic inference for richly structured spatial data. The term “full” loosely refers to the estimation of *all* unknowns in the model, including all parameters (irrespective of how well they are informed by the data), random effects, and predictive random variables for probabilistic interpolation at arbitrary locations (and time points in spatial-temporal data).

Spatial data analysis employs Gaussian processes (GPs) to model dependence and achieves superior predictive inference. While GPs offer flexibility and are a conspicuous choice in spatial modeling, they generally do not offer computationally exploitable structures for covariance matrices. The computational requirements for full inference become onerous and impractical for large datasets. This presents a conundrum: should we strive to completely retain the probabilistic rigors of theoretical inference, which may be computationally burdensome or even impracticable given the massive amounts of data, or should we concede uncertainty quantification and adopt algorithm-driven data analysis that would easily scale massive data sets? Statistical learning on unprecedented scales requires some concessions from conventional decision-theoretic paradigms, as has been reflected, for example, in a recent comprehensive case study by Zhou et al. (2022), who synthesize statistical models with machine learning algorithms to measure temporal trends and spatial distribution of housing vitality by exploiting information from multiple sources (also see the discussion by Banerjee, 2022, for conventional Bayesian modeling perspectives and its challenges). Other efforts in combining machine learning methods with formal spatial statistics include deep learning for spatial data (Zammit-Mangion et al., 2022; Wikle and Zammit-Mangion, 2023, with the latter offering a comprehensive review of spatial machine learning) and, in a related domain, deep GP emulation in computer experiments (Sauer et al., 2023b), and spatial random forests (Georganos et al., 2021; Talebi et al., 2022; Saha et al., 2023).

Even a cursory review reveals various methods for massive spatial datasets, which are too vast to be summarized here (see, e.g., Banerjee, 2017; Heaton et al., 2017). Examples include reduced-rank or subsets of regression approaches (see, e.g., Quiñonero-Candela and Rasmussen, 2005; Cressie and Johannesson, 2008; Banerjee et al., 2008; Wikle, 2010), multi-resolution approaches (Nychka et al., 2015; Katzfuss, 2017), and graph-based models (Vecchia, 1988; Datta et al., 2016; Katzfuss and Guinness, 2021; Peruzzi et al., 2022; Dey et al., 2022; Sauer et al., 2023a). Full inference typically requires Markov chain Monte Carlo (MCMC) (Finley et al., 2019), variational approximations (Ren et al., 2011; Wu et al., 2022; Cao et al., 2023) or Gaussian Markov random field approximations (Rue et al., 2009; Lindgren et al., 2011) and integrated nested Laplace approximations (INLA) for computing the marginal posterior distributions of the process.

The aforementioned methods focus on the richness of statistical inference but can be encumbered by the challenges presented by spatial data analysis. To be specific, we seek inference for the spatial process, which is an uncountable set of random variables, based only on a partial realization of the process manifested in the form of measurements at a finite set of spatial locations (see, e.g., Chapter 3 in Banerjee et al., 2025b). In particular, a careful choice of prior distributions and subsequent tuning of the iterative estimation algorithms are needed because spatial covariance kernel parameters are typically not statistically identifiable or consistently estimable from the observed data (Zhang, 2004; Tang et al., 2021). Exploratory data analysis (EDA) methods (such as the variogram analysis presented in Appendix D of this article; also see, e.g., Chapter 2 in Banerjee et al., 2025b, for details on spatial EDA) provide some insights into the values of spatial covariance kernel

parameters generating the data. Geostatistical models can then be fixed over a finite set of values for these parameters, and the inference can be assimilated or “stacked” over these models (Zhang et al., 2025; Pan et al., 2025).

Our approach relies on two basic tenets: (i) model-based statistical inference for underlying spatial processes (including multivariate processes) in a robust and largely automated manner with minimal human input; and (ii) achieving such inference for truly massive amounts of data without resorting to iterative algorithms that may require significant human intervention to diagnose convergence (such as in MCMC).

We retain the benefits of posterior learning of the underlying stochastic process. In particular, we obtain predictive learning over the uncountable collection of points in the spatial domain while propagating uncertainty about the model’s parameters. However, some compromises regarding the richness of models are inevitable from the viewpoint of GEOAI. At the same time, statistical distribution theory has much to offer, and we harness analytical closed-form distribution theory within a family of matrix-variate distributions to deliver inference. This is only possible by “fixing” certain spatial correlation kernel parameters that are weakly identified and hinder automated inference. Hence, we infer using closed-form matrix-variate distributions corresponding to a set of fixed values of hyperparameters and subsequently assimilate posterior learning using Bayesian predictive stacking or BPS (Wolpert, 1992; Yao et al., 2018; Zhang et al., 2025; Pan et al., 2025). Stacking (Breiman, 1996) is primarily related to predictions by minimizing the generalization error rate of one or more models. Bayesian predictive stacking (BPS) is a particular type of Bayesian model averaging and, more broadly, Bayesian predictive synthesis (see, e.g., Clarke, 2003; Clyde and Iversen, 2013; Le and Clarke, 2017; McAlinn and West, 2019; McAlinn et al., 2020; Yao et al., 2022; Tallman and West, 2023; Cabel et al., 2025, for different perspectives).

A GEOAI system will access massive amounts of data. We devise Bayesian transfer learning on a streaming set of spatial datasets. Data flows into the GEOAI system in the form of K subsets, each with $\approx n/K$ locations. We analyze and propagate inference from one subset to the next and assimilate inferences from these subsets. Delegating Bayesian calculations to a group of independent datasets is intuitive and has been explored from diverse perspectives. Examples include Consensus Monte Carlo (CMC) (Scott et al., 2016) and its kernel-based adaptations (Rendell et al., 2021), Bayesian meta-analysis in clinical applications (see, e.g., Chapter 4 in Parmigiani, 2002), and in spatially-temporally structured data (Bell et al., 2005; Kang et al., 2011); and more recently, “meta” approaches using diverse distributed computing architectures (Srivastava et al., 2015; Deisenroth and Ng, 2015; Minsker et al., 2017; Srivastava et al., 2018; Guhaniyogi and Banerjee, 2018, 2019; Guhaniyogi et al., 2019, 2022, 2023). Such approaches have typically relied on iterative algorithms to estimate weakly identifiable parameters. Instead, we propose transfer learning using a double stacking approach. We first obtain stacked posterior distributions within each subset and then assimilate inference across subsets by a second stacking algorithm. In Figure 1, a flow chart is provided comprising three steps: (i) partition the data into smaller subsets; (ii) analyze each subset by stacking conjugate posteriors in closed form; and (iii) stack across datasets to obtain inference for all data.

This article evolves according to Figure 1. Section 2 outlines Bayesian transfer learning using DOUBLE BPS (DBPS), Section 4 provides details on computation, Section 5 presents simulation experiments to illustrate and evaluate DOUBLE BPS and its applicability to amor-

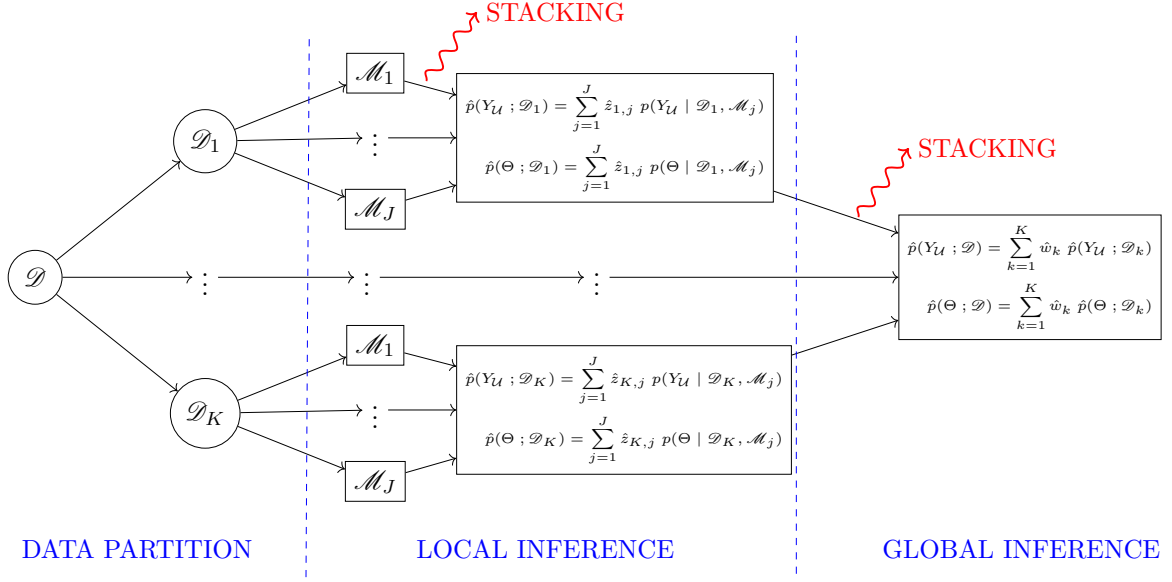


Figure 1: Double Bayesian predictive stacking approach representation

tized inference. Section 6 analyzes a vegetation index data set with observed locations in the millions on global scales. Finally, Section 7 concludes with some discussion. References to an accompanying Supplement are provided throughout the article and after Section 7.

2. Bayesian Transfer Learning

Transfer learning (TL) broadly refers to the propagation of knowledge from one task to accomplish a different task (Suder et al., 2025). We transfer inference through a stream of subsets to analyze the entire dataset. Learning from spatial random fields is challenging because of the limited information available for the latent spatial process (uncountable collection of unobserved random variables) using a partially observed finite realization. We devise an automated approach for amortized learning (Zammit-Mangion et al., 2025).

2.1 Divide-and-Conquer Multivariate Bayesian inference

Let $Y_{n \times q}$ be an $n \times q$ random matrix that is endowed with a probability law from the matrix-normal distribution, $\text{MN}(M, V, U)$, with a probability density function

$$p(Y | M, V, U) = \frac{\exp \left[-\frac{1}{2} \text{tr} \{ U^{-1} (Y - M)^T V^{-1} (Y - M) \} \right]}{(2\pi)^{\frac{nq}{2}} |U|^{\frac{n}{2}} |V|^{\frac{q}{2}}}, \quad (1)$$

where $\text{tr}(\cdot)$ is the trace operator on a square matrix, M is the mean matrix, and V and U are the $n \times n$ row-covariance and $q \times q$ column covariance matrices, respectively. We consider the matrix-variate Bayesian linear regression model

$$Y = X\beta + E_Y, \quad E_Y \sim \text{MN}(O, V, \Sigma); \quad \beta = M_0 m_0 + E_\beta, \quad E_\beta \sim \text{MN}(O, M_0, \Sigma), \quad (2)$$

where Y is $n \times q$, X is $n \times p$ comprising explanatory variables, β is $p \times q$ consisting of regression slopes, E_Y and E_β are zero-centered random matrices with row covariances V and M_0 , respectively, and a shared column covariance matrix Σ . We assign an inverse-Wishart distribution $\Sigma \sim \text{IW}(\Psi_0, \nu_0)$ and denote the joint density of β and Σ by $\text{MNIW}(M_0 m_0, M_0, \Psi_0, \nu_0)$, which provides a closed-form posterior for $\{\beta, \Sigma\}$ in the same family.

Let $\mathcal{D} = \{Y, X\}$ be the entire dataset, which is too large to even be accessed, let alone analyzed using (1) within the GEOAI system. Therefore, we envisage K disjoint and exhaustive subsets $\mathcal{D} = \{\mathcal{D}_1, \dots, \mathcal{D}_K\}$ streaming into GEOAI as a sequence. Each $\mathcal{D}_k = \{Y_k, X_k\}$ consists of n_k rows of Y and X , where $n = \sum_{k=1}^K n_k$, Y_k is $n_k \times q$ and X_k is $n_k \times p$ for each $k = 1, \dots, K$. We now fit (2) to each subset using $Y_k = X_k \beta + E_k$, $E_k \sim \text{MN}(O, V_k, \Sigma)$, where V_k is the $n_k \times n_k$ row-covariance matrix corresponding to the rows in Y_k ; the specification for $\{\beta, \Sigma\}$ remains as in (2). Starting with $\text{MNIW}(\beta, \Sigma \mid M_k m_k, M_k, \Psi_k, \nu_k)$ at $k = 0$ (the prior), we use Bayesian updating $p(\beta, \Sigma \mid \mathcal{D}_{1:k+1}) \propto p(\beta, \Sigma \mid \mathcal{D}_{1:k}) \times p(Y_{k+1} \mid X_{k+1} \beta, V_{k+1}, \Sigma)$ to obtain $\beta, \Sigma \mid \mathcal{D}_{1:k+1} \sim \text{MNIW}(M_{k+1} m_{k+1}, M_{k+1}, \Psi_{k+1}, \nu_{k+1})$ with $M_{k+1}^{-1} = M_k^{-1} + X_{k+1}^\top V_{k+1}^{-1} X_{k+1}$, $m_{k+1} = m_k + X_{k+1}^\top V_{k+1}^{-1} Y_{k+1}$, $\nu_{k+1} = \nu_k + n_{k+1}$, and $\Psi_{k+1} = \Psi_k + Y_{k+1}^\top V_{k+1}^{-1} Y_{k+1} + m_k^\top M_k m_k - m_{k+1}^\top M_{k+1} m_{k+1}$. Upon terminating at $k = K$, we exactly recover the posterior distribution $p(\beta, \Sigma \mid \mathcal{D})$. We do not interact between subsets, and computational complexity is determined solely by the dimension of the subsets.

Spatial random fields immediately present a challenge. The above method delivers inference without loss of information only if the Y_k 's are independent between blocks. If each row of Y corresponds to a spatial location, and V is an $n \times n$ spatial correlation matrix, then each V_k is the spatial correlation matrix constructed from spatial locations n_k in $\mathcal{D}_k$. The independence among blocks may yield reasonable inference if we can design the blocks such that spatial correlation between blocks does not affect inference (see Section 4 of Banerjee et al., 2025a, for a ‘‘seasons and episodes’’ framework to adapt the Markovian Forward Filter Backward Sampling (FFBS) algorithm for spatial-temporal learning). However, designing such blocks requires significant human intervention. We seek to avoid this in transfer learning and, in particular, in amortized inference, which will require rapid estimation of datasets to train deep networks (see Section 5.3).

Instead, we assimilate statistical learning from each of these blocks using predictive stacking. We exploit the fact that V is indexed by a small number of parameters in a spatial correlation kernel. Fixing these parameters fixes V , and hence V_k for each k , producing closed-form posterior inference on β and Σ based on the entire dataset as described above. Stacking combines these analytically accessible distributions using an optimal set of weights that are computed using a convex optimization algorithm. These weights are then used to reconstruct the posterior and predictive distributions for the spatial random field without imposing block independence. Existing ‘‘meta-kriging’’ and related ‘‘divide and conquer’’ approaches (e.g., Guhaniyogi and Banerjee, 2018; Scott et al., 2016, and other references on ‘‘divide and conquer’’ methods provided in Section 1) analyze subsets of data using MCMC, which is expensive and not fully automated. A key distinction of the current manuscript is that we abandon all iterative estimation algorithms, let alone expensive MCMC, and focus on assimilating inference using closed form distributions.

What remains to be resolved is the issue of fixing the parameters in V . These parameters govern the strength of association across the spatial random field and possibly the smoothness of the field. Unfortunately, these parameters are weakly identified by the data,

and posterior learning struggles due to the slower convergence of iterative algorithms. In addition, exploratory spatial data science tools, such as variograms, used to glean information about these parameters will also be less helpful, as they may suggest different values for each of the variables indexed by columns, while we insist on retaining a single parameter to exploit conjugate distribution theory. Hence, we collect the closed-form posterior distributions obtained for a collection of fixed values of the spatial parameters and subsequently average these posterior distributions.

2.2 Bayesian Stacking of Predictive Densities

Bayesian predictive stacking (BPS) assimilates models using a weighted distribution in the convex hull, $C = \left\{ \sum_{j=1}^J w_j p(\cdot | \mathcal{D}, \mathcal{M}_j) : \sum_j w_j = 1, w_j \geq 0 \right\}$, of individual posterior distributions by maximizing the score (Gneiting and Raftery, 2007; Yao et al., 2018) to fetch

$$(w_1, \dots, w_J)^T = \arg \max_{w \in S_1^J} \frac{1}{n} \sum_{i=1}^n \log \sum_{j=1}^J w_j p(Y_i | \mathcal{D}_{-i}, \mathcal{M}_j) , \quad (3)$$

where $\mathcal{D}_{-i}$ are the data that exclude the i -th block (indexed by a row) of observations in Y , and $\mathcal{M} = (\mathcal{M}_1, \dots, \mathcal{M}_J)$ are J different models. For any given data set $\mathcal{D}$, these J different models correspond to fixed spatial correlation kernel parameters in V . Solving (3) minimizes the Kullback-Leibler divergence from the true predictive distribution by using convex optimization (Grant, 2005; CVX Research, 2012). Although the true predictive distribution is unknown, we target a leave-one-out (LOO) estimate of the expected value of the score (see. e.g., Yao et al., 2018, for details) as the theoretical criterion for computing stacking weights. Since exact LOO requires fitting the model n times, as we exclude one row of data $\mathcal{D}$ at a time, we employ K -fold cross-validation as a more cost-effective and tractable method for generating predictions (Breiman, 1996).

2.3 Accelerated Learning for Spatial Random Fields

Let $\mathcal{S} = \{s_1, \dots, s_n\} \subset \mathcal{D}$ be a set of n locations that produce observations on q possibly correlated outcomes collected into a $q \times 1$ vector $y(s) = (y_1(s), \dots, y_q(s))^T$ for each $s \in \mathcal{S}$. We collect these measurements in the matrix $n \times q$ $Y = [y_j(s_i)^T]$ for $i = 1, \dots, n$ and $j = 1, \dots, q$. Let $X = [x(s_i)^T]$ be $n \times p$ with rows $x(s_i)^T$ consisting of $p < n$ explanatory variables at location $s_i \in \mathcal{S}$; we assume that X has rank p . We introduce latent spatial processes, $\omega_j(s)$, for each outcome $y_j(s)$ to capture spatial dependence and a $q \times q$ covariance matrix, Σ , to capture non-spatial dependence among the elements of $y(s)$ within s . This matrix is typically adjusted by a scale factor $(\alpha^{-1} - 1)$ to accommodate additional variation on local scales. Thus, setting $\alpha = \sigma^2 / (\sigma^2 + \tau^2)$ where σ^2 and τ^2 denote variances for the spatial process and measurement error (“nugget”) implies α is the ratio of the spatial variance (partial sill) to the total variance (sill) gleaned from a variogram.

We cast this into (2), but we explicitly introduce a latent $q \times 1$ spatial process $\omega(s)$ as

$$\begin{aligned} Y &= X\beta + \Omega + E, \quad E | \Sigma \sim \text{MN}(O, (\alpha^{-1} - 1)\mathbb{I}_n, \Sigma); \quad \Sigma \sim \text{IW}(\Psi_0, \nu_0); \\ \beta &= M_0 m_0 + E_\beta, \quad E_\beta | \Sigma \sim \text{MN}(O, M_0, \Sigma); \quad \Omega | \Sigma \sim \text{MN}(O, V, \Sigma), \end{aligned} \quad (4)$$

where $\Omega = [\omega(s_i)^T]$ is $n \times q$ with rows $\omega(s_i)^T$. To capture spatial dependence, V is an $n \times n$ spatial correlation matrix with the (i, j) -th element equal to the value of a positive definite spatial correlation function $\rho(s_i, s_j; \phi)$ indexed by the parameter(s) ϕ . We account for measurement errors in observations so that the spatial component of the variation in the elements of Y is modeled using $\rho(s_i, s_j; \phi) + (\alpha^{-1} - 1)\mathbb{1}_{s_i=s_j}$, where $\alpha \in [0, 1]$ is the proportion of total variability attributed to the spatial process.

Letting $\gamma^T = [\beta^T, \Omega^T]$ be $q \times (p + n)$, we assume $\{\gamma, \Sigma\} \sim \text{MNIW}(\mu_\gamma, V_\gamma, \Psi_0, \nu_0)$, where

$$\text{MNIW}(\gamma, \Sigma \mid \mu_\gamma, V_\gamma, \Psi_0, \nu_0) = \text{IW}(\Sigma \mid \Psi_0, \nu_0) \times \text{MN}_{p+n, q}(\gamma \mid \mu_\gamma, V_\gamma, \Sigma), \quad (5)$$

with $\mu_\gamma^T = [m_0^T M_0, 0_{q \times n}]$ and $V_\gamma = \text{blockdiag}\{M_0, \rho_\phi(\mathcal{S}, \mathcal{S})\}$. The MNIW prior is conjugate with respect to the matrix-normal likelihood. Thus, for any fixed $\{\alpha, \phi\}$ and hyperparameters in the prior density, we obtain an MNIW posterior density for $\{\gamma, \Sigma\}$,

$$p(\gamma, \Sigma \mid \mathcal{D}) = \text{MNIW}(\gamma, \Sigma \mid \mu_\gamma^*, V_\gamma^*, \Psi^*, \nu^*), \quad (6)$$

where $V_\gamma^* = \begin{bmatrix} \frac{\alpha}{1-\alpha} X^T X + M_0^{-1} & \frac{\alpha}{1-\alpha} X^T \\ \frac{\alpha}{1-\alpha} X & \rho_\phi^{-1}(\mathcal{S}, \mathcal{S}) + \frac{\alpha}{1-\alpha} \mathbb{I}_n \end{bmatrix}^{-1}$ and $\mu_\gamma^* = V_\gamma^* \begin{bmatrix} \frac{\alpha}{1-\alpha} X^T Y + m_0 \\ \frac{\alpha}{1-\alpha} Y \end{bmatrix}$, $\Psi^* = \Psi_0 + \frac{\alpha}{1-\alpha} Y^T Y + m_0^T M_0 m_0 - \mu_\gamma^{*T} V_\gamma^{*-1} \mu_\gamma^*$ and $\nu^* = \nu_0 + n$.

The framework in (4) is equivalent to (2) with $V = R_\phi + (\alpha^{-1} - 1)\mathbb{I}_n$ and $R_\phi = [\rho(s_i, s_j; \phi)]$. We recover posterior samples of Ω by using each posterior draw of $\{\beta, \Sigma\}$ to draw a corresponding Ω from $p(\Omega \mid \mathcal{D}, \beta, \Sigma, \mathcal{M}_j)$. This seamlessly renders itself to the Bayesian transfer learning framework described in Section 2.1, provided that V or $\{\alpha, \phi\}$ is fixed. For GEOAI, we seek to minimize human intervention. Rather than fixing them at one particular value, perhaps gleaned from a spatial variogram that requires human inspection, we use a set of J candidate values $\{\alpha_j, \phi_j\}$ specifying model $\mathcal{M}_j$ for $j = 1, \dots, J$. We now obtain analytical closed forms for $p(\beta, \Sigma \mid \mathcal{D}, \mathcal{M}_j)$ for each j , as described in Section 2.1, and use Bayesian predictive stacking to evaluate the stacked posterior distribution.

Turning to prediction, let $\mathcal{U} = \{u_1, \dots, u_{n'}\}$ be a finite set of locations where we seek to predict or impute the value of Y based on an observed $n' \times p$ design matrix $X_{\mathcal{U}}$ associated with the locations in $\mathcal{U}$. The joint posterior predictive for $Y_{\mathcal{U}}$ and the unobserved latent process $\Omega_{\mathcal{U}} = [\omega(u_i)^T]$ for $i = 1, \dots, n'$ can be recast by integrating out $\{\gamma, \Sigma\}$ from the conditional posterior predictive distribution to yield

$$p(Y_{\mathcal{U}}, \Omega_{\mathcal{U}} \mid \mathcal{D}) = \int \text{MN}_{n', q}(Y_{\mathcal{U}} \mid X_{\mathcal{U}}\beta + \Omega_{\mathcal{U}}, (\alpha^{-1} - 1)\mathbb{I}_{n'}, \Sigma) \times \text{MN}_{n', q}(\Omega_{\mathcal{U}} \mid M_{\mathcal{U}}\Omega, V_{\Omega_{\mathcal{U}}}, \Sigma) \times \text{MNIW}(\gamma, \Sigma \mid \mu_\gamma^*, V_\gamma^*, \Psi^*, \nu^*) d\gamma d\Sigma, \quad (7)$$

where $M_{\mathcal{U}} = \rho_\phi(\mathcal{U}, \mathcal{S})\rho_\phi^{-1}(\mathcal{S}, \mathcal{S})$ and $V_{\Omega_{\mathcal{U}}} = \rho_\phi(\mathcal{U}, \mathcal{U}) - \rho_\phi(\mathcal{U}, \mathcal{S})\rho_\phi^{-1}(\mathcal{S}, \mathcal{S})\rho_\phi(\mathcal{S}, \mathcal{U})$. This is a matrix-variate t distribution $\text{T}_{2n', q}(\nu^*, \mu^*, V^*, \Psi^*)$, with degrees of freedom ν^* , location matrix $\mu^* = M\mu_\gamma^*$, row-scale matrix V^* , and column-scale matrix Ψ^* , where $M = \begin{bmatrix} 0 & M_{\mathcal{U}} \\ X_{\mathcal{U}} & M_{\mathcal{U}} \end{bmatrix}$ and $V^* = MV_\gamma^*M^T + V_E$ with $V_E = \begin{bmatrix} V_{\Omega_{\mathcal{U}}} & V_{\Omega_{\mathcal{U}}} \\ V_{\Omega_{\mathcal{U}}} & V_{\Omega_{\mathcal{U}}} + (\alpha^{-1} - 1)\mathbb{I}_{n'} \end{bmatrix}$. See Section A.1 for details. While the conditional posterior predictive distributions take the following forms $p(\Omega_{\mathcal{U}} \mid \mathcal{D}, \gamma, \Sigma) = \text{MN}_{n', q}(\Omega_{\mathcal{U}} \mid M_{\mathcal{U}}\Omega, V_{\Omega_{\mathcal{U}}}, \Sigma)$ and $p(Y_{\mathcal{U}} \mid \mathcal{D}, \Omega_{\mathcal{U}}, \gamma, \Sigma) = \text{MN}_{n', q}(Y_{\mathcal{U}} \mid X_{\mathcal{U}}\beta + \Omega_{\mathcal{U}}, (\alpha^{-1} - 1)\mathbb{I}_{n'}, \Sigma)$. Hence, we can proceed with posterior predictive inference

Algorithm 1 Computing stacking weights within subsets using BPS

Input: Y ($n \times q$ matrix of outcomes), X ($n \times p$ design matrix), $\mathcal{S}$ (coordinates of n locations); $\{m_0, M_0, \Psi_0, \nu_0\}$: Prior parameters; $G_\alpha \times G_\phi$: Grids of $\{\alpha, \phi\}$; n (no. of locations), q (no. of outcomes), p (no. of predictors); K (no. of subsets), J (no. of models), L (no. of folds).
Output: $\hat{z} = \{\hat{z}_k = \{\hat{z}_{k,j}\} : k = 1, \dots, K, j = 1, \dots, J\}$: Stacking weights within subsets; $\{pd_{k,j,i} : k = 1, \dots, K, j = 1, \dots, J, i = 1, \dots, n\}$: point-wise predictive density of Y ; G_{all} : Grid of dimension J , spanned by G_α, G_ϕ

- 1: Partition $Y, X, \mathcal{S}$ into $\mathcal{D}_k = \{Y_k, X_k, \mathcal{S}_k\}, k = 1, \dots, K$
- 2: Store n_k , as cardinality of $\mathcal{S}_k$; Compute G_{all} by expanding G_α, G_ϕ
- 3: **for** $k = 1, \dots, K$ **do Parallel**
- 4: **for** $j = 1, \dots, J$ **do**
- 5: Extract $\{\alpha_j, \phi_j\}$ from j -th row of G_{all}
- 6: Form L folds: $\mathcal{D}_{k,[l]} = \{Y_{k,[l]}, X_{k,[l]}, \mathcal{S}_{k,[l]}\}$ and $\mathcal{D}_{k,[-l]} = \{Y_{k,[-l]}, X_{k,[-l]}, \mathcal{S}_{k,[-l]}\}$
- 7: Store $n_{k,[-l]}$, as cardinality of $\mathcal{S}_{k,[-l]}$
- 8: **for** $l = 1, \dots, L$ **do**
- 9: Compute $R_{\phi_j}([-l]) = \rho_{\phi_j}(\mathcal{S}_{k,[-l]}, \mathcal{S}_{k,[-l]}), R_{\phi_j}^{-1}([-l])$ and M_0 for M_0^{-1}
- 10: Construct $V_{\gamma,[-l]}^{\star-1} = \begin{bmatrix} \frac{\alpha_j}{1-\alpha_j} X_{k,[-l]}^T X_{k,[-l]} + M_0^{-1} & \frac{\alpha_j}{1-\alpha_j} X_{k,[-l]}^T \\ \frac{\alpha_j}{1-\alpha_j} X_{k,[-l]} & R_{\phi_j}^{-1}([-l]) + \frac{\alpha_j}{1-\alpha_j} \mathbb{I}_{n_{k,[-l]}} \end{bmatrix}$
- 11: Solve for $\mu_{\gamma,[-l]}^*$: $V_{\gamma,[-l]}^{\star-1} \mu_{\gamma,[-l]}^* = \begin{bmatrix} \frac{\alpha_j}{1-\alpha_j} X_{k,[-l]}^T Y_{k,[-l]} + m_0 \\ \frac{\alpha_j}{1-\alpha_j} Y_{k,[-l]} \end{bmatrix}$
- 12: Calculate $\Psi_{[-l]}^* = \Psi_0 + (\alpha_j^{-1} - 1)^{-1} Y_{k,[-l]}^T Y_{k,[-l]} + m_0^T M_0 m_0 - \mu_{\gamma,[-l]}^{T*} V_{[-l]}^{\star-1} \mu_{\gamma,[-l]}^*$
- 13: Calculate $\nu_{[-l]}^* = \nu_0 + n_{k,[-l]}$
- 14: **for** $i \in [l]$ **do**
- 15: Compute $R_{\phi_j}(i) = \rho_{\phi_j}(\mathcal{S}_{k,i}, \mathcal{S}_{k,i})$ and $R_{\phi_j}(i, [-l]) = \rho_{\phi_j}(\mathcal{S}_{k,i}, \mathcal{S}_{k,[-l]})$
- 16: Calculate $M_i = R_{\phi_j}(i, [-l]) R_{\phi_j}^{-1}([-l])$ and form $M_{y,i} = [X_{k,i} \quad M_i]$
- 17: Calculate $\mu_i^* = M_{y,i} \mu_{\gamma,[-l]}^*$ and $V_{\Omega_i} = R_{\phi_j}(i) - M_i R_{\phi_j}([-l], i)$
- 18: Construct $V_{e,i} = V_{\Omega_i} + (\alpha_j^{-1} - 1)$ and $V_i^* = M_{y,i} V_{\gamma,[-l]}^* M_{y,i}^T + V_{e,i}$
- 19: Compute $pd_{k,j,i} = T_{1,q}(Y_{k,i} \mid \nu_{[-l]}^*, \mu_i^*, V_i^*, \Psi_{[-l]}^*)$.
- 20: **end for**
- 21: **end for**
- 22: **end for**
- 23: Solve: $\max_{z_k \in S_1^J} \frac{1}{n_k} \sum_{i=1}^{n_k} \log \sum_{j=1}^J z_{k,j} pd_{k,j,i}$ such that $z_k \in [0, 1]^J : \sum_{j=1}^J z_{k,j} = 1$
- 24: **end Parallel for**
- 25: **return** $\{\hat{z}, \{pd_{k,j,i}\}, G_{all}\}$

by sampling from the closed-form joint predictive distribution or sampling from the conditional distributions. We draw one instance of $\Omega_{\mathcal{U}} \sim p(\Omega_{\mathcal{U}} \mid \mathcal{D}, \gamma, \Sigma)$ for each posterior draw of $\{\gamma, \Sigma\}$ and then draw a matrix $Y_{\mathcal{U}}$ from $p(Y_{\mathcal{U}} \mid \mathcal{D}, \Omega_{\mathcal{U}}, \gamma, \Sigma)$ for each drawn $\{\Omega_{\mathcal{U}}, \gamma, \Sigma\}$. The resulting samples are exactly drawn from the posterior predictive distribution $p(Y_{\mathcal{U}} \mid \mathcal{D})$.

This tractability is possible if the range decay ϕ and α are fixed. While data can inform about these parameters, they are inconsistently estimable and lead to poorer convergence (Zhang, 2004). Finley et al. (2019) explored K -fold cross-validation, but inference is limited

to only one set of values for the parameters. Instead, we pursue exact inference using (6) and (7) by stacking over different fixed values of $\{\alpha, \phi\}$ using BPS of predictive densities as described in Section 2.2. This minimizes human intervention and enables automation.

For each subset of the data, we compute the stacking weights $z_k = \{z_{k,j}\}_{j=1,\dots,J}$ as

$$\max_{z_k \in S_1^J} \frac{1}{n_k} \sum_{i=1}^{n_k} \log \sum_{j=1}^J z_{k,j} p(Y_{k,i} \mid \mathcal{D}_{k,[-l]}, \mathcal{M}_j), \quad (8)$$

where $Y_{k,i}$ is the i -th of the n_k rows of $Y_{k,[l]} \in \mathcal{D}_{k,[l]}$ (the l -th fold within the k -th dataset), and $\mathcal{D}_{k,[-l]}$ is the k -th dataset without the l -th fold, with $l = 1, \dots, L$, and L the number of folds for K-fold cross-validation estimates for the expected value of the score (see Section 2.2). Analogously, $X_{k,i}$ denotes the i -th row of $X_{k,[l]}$ and $\mathcal{S}_{k,i}$ contains the associated location, so that $\{Y_{k,i}, X_{k,i}, \mathcal{S}_{k,i}\}$ collects the i -th held-out observation in the l -th fold. The posterior predictive density, $p(Y_{k,i} \mid \mathcal{D}_{k,[-l]}, \mathcal{M}_j)$, is available in closed form as a matrix t distribution, which makes the computation efficient. This leads us from (8) to

$$\max_{z_k \in S_1^J} \frac{1}{n_k} \sum_{i=1}^{n_k} \log \sum_{j=1}^J z_{k,j} \text{T}_{1,q}(Y_{k,i} \mid \nu_{[-l]}^*, \mu_i^*, V_i^*, \Psi_{[-l]}^*), \quad (9)$$

where $\nu_{[-l]}^* = \nu_0 + n_{k,[-l]}$, $n_{k,[-l]}$ is the cardinality of $\mathcal{S}_{k,[-l]}$ (which is the set of locations in $\mathcal{D}_{k,[-l]}$), and $\Psi_{[-l]}^* = \Psi_0 + (\alpha_j^{-1} - 1)^{-1} Y_{k,[-l]}^T Y_{k,[-l]} + m_0^T M_0 m_0 - \mu_{\gamma,[-l]}^{T*} V_{\gamma,[-l]}^{*-1} \mu_{\gamma,[-l]}^*$. The quantities $V_i^* = M_{y,i} V_{\gamma}^* M_{y,i}^T + V_{\Omega_i} + (\alpha_j^{-1} - 1)$ and $\mu_i^* = M_{y,i} \mu_{\gamma,[-l]}^*$ are computed from the fol-

lowing auxiliary quantities: $V_{\gamma,[-l]}^{*-1} = \begin{bmatrix} \frac{\alpha_j}{1-\alpha_j} X_{k,[-l]}^T X_{k,[-l]} + M_0^{-1} & \frac{\alpha_j}{1-\alpha_j} X_{k,[-l]}^T \\ \frac{\alpha_j}{1-\alpha_j} X_{k,[-l]} & \rho_{\phi_j}^{-1}(\mathcal{S}_{k,[-l]}, \mathcal{S}_{k,[-l]}) + \frac{\alpha_j}{1-\alpha_j} \mathbb{I}_{n_{k,[-l]}} \end{bmatrix}$,

$\mu_{\gamma,[-l]}^* = V_{\gamma,[-l]}^* \begin{bmatrix} \frac{\alpha_j}{1-\alpha_j} X_{k,[-l]}^T Y_{k,[-l]} + m_0 \\ \frac{\alpha_j}{1-\alpha_j} Y_{k,[-l]} \end{bmatrix}$, $M_{y,i} = [X_{k,i} \quad \rho_{\phi_j}(\mathcal{S}_{k,i}, \mathcal{S}_{k,[-l]}) \rho_{\phi_j}^{-1}(\mathcal{S}_{k,[-l]}, \mathcal{S}_{k,[-l]})]$, and

$V_{\Omega_i} = \rho_{\phi_j}(\mathcal{S}_{k,i}, \mathcal{S}_{k,i}) - \rho_{\phi_j}(\mathcal{S}_{k,i}, \mathcal{S}_{k,[-l]}) \rho_{\phi_j}^{-1}(\mathcal{S}_{k,[-l]}, \mathcal{S}_{k,[-l]}) \rho_{\phi_j}(\mathcal{S}_{k,[-l]}, \mathcal{S}_{k,i})$. Note that $\nu_{[-l]}^*$ is scalar, μ_i^* is $(1 \times q)$ row vector, V_i^* is a scalar, and $\Psi_{[-l]}^*$ is $(q \times q)$ matrix, so that (9) evaluates the density of a $(1 \times q)$ matrix-variate t distribution. Further details, including derivations and implementation, are supplied in Section A.1 and Algorithm 1.

For each dataset $\mathcal{D}_k$, BPS computes: (i) an estimate of the posterior predictive $\hat{p}(\cdot; \mathcal{D}_k) = \sum_{j=1}^J \hat{z}_{k,j} p(\cdot \mid \mathcal{D}_k, \mathcal{M}_j)$ for $k = 1, \dots, K$ and $j = 1, \dots, J$; and (ii) a set of stacking weights $\hat{z}_k = \{\hat{z}_{k,j}\}_{j=1,\dots,J}$. Once these weights are available, we apply BPS a second time to obtain a weighted average of $\hat{p}(\cdot; \mathcal{D}_k)$ over the k subsets. This DOUBLE BPS (DBPS) of predictive densities seeks weights $w = \{w_k\}_{k=1,\dots,K}$ such that $\hat{w} = \max_{w \in S_1^K} \frac{1}{n} \sum_{i=1}^n \log \sum_{k=1}^K w_k \hat{p}(Y_i; \mathcal{D}_k)$ (see Section B.1). Once stacking weights $\hat{w} = \{\hat{w}_k\}_{k=1,\dots,K}$ are computed using Algorithm 2, sampling from the posterior and posterior predictive distributions is obtained from

$$\hat{p}(\cdot; \mathcal{D}) = \sum_{k=1}^K \hat{w}_k \sum_{j=1}^J \hat{z}_{k,j} p(\cdot \mid \mathcal{D}_k, \mathcal{M}_j). \quad (10)$$

Only posterior predictive distributions are considered to acquire the two sets of stacking weights. Given the weights obtained from double stacking, the stacked posterior distribution is a mixture of finite mixtures. This makes sampling from (10) straightforward.

Algorithm 2 Calculating stacking weights between subsets using BPS

Input: $\hat{z} = \{\hat{z}_k = \{\hat{z}_{k,j}\} : k = 1, \dots, K, j = 1, \dots, J\}$: Stacking weights within subsets; $\{pd_{k,j,i} : k = 1, \dots, K, j = 1, \dots, J, i = 1, \dots, n\}$: point-wise predictive density of Y ; n, q, p : Number of rows, number of outcomes, and number of predictors; $K, \{n_k : k = 1, \dots, K\}, J$: Number of subsets, dimension of each subset, and number of competitive models in each subset.

Output: $\hat{w} = \{\hat{w}_k : k = 1, \dots, K\}$: Stacking weights between subsets.

- 1: Construct $pd = [pd_1^T : \dots : pd_K^T]^T$ of dimension $(n \times J)$
 where $pd_k = \begin{bmatrix} pd_{k,1,1} & \dots & pd_{k,J,1} \\ \vdots & & \vdots \\ pd_{k,1,n_k} & \dots & pd_{k,J,n_k} \end{bmatrix}$ of dimension $(n_k \times J)$
- 2: **for** $k = 1, \dots, K$ **do**
- 3: Compute $epd_k = pd \hat{z}_k$ of dimension $(n \times 1)$
- 4: **end for**
- 5: Solve convex optimization problem:

$$\max_{w \in S_1^K} \frac{1}{n} \sum_{i=1}^n \log \sum_{k=1}^K w_k epd_{k,i} = \max_{w \in S_1^K} \frac{1}{n} \sum_{i=1}^n \log \sum_{k=1}^K w_k \sum_{j=1}^J \hat{z}_{k,j} pd_{k,j,i}$$
 where $pd_{k,j,i} = p(Y_{k,i} \mid \mathcal{D}_{k,[-l]}, \mathcal{M}_j) = T_{1,q}(Y_{k,i} \mid \nu_{[-l]}^*, \mu_i^*, V_i^*, \Psi_{[-l]}^*)$
 for $\forall i \notin [-l], l \in \{1, \dots, L\}$ and $S_1^K = \{w \in [0, 1]^K : \sum_{k=1}^K w_k = 1\}$
- 6: **return** $\hat{w} = \{\hat{w}_k : k = 1, \dots, K\}$

First, the set of stacking weights $\hat{z}_k = \{\hat{z}_{k,j}\}_{j=1,\dots,J}$ obtained using BPS within the subset of the data $\mathcal{D}_k$ is primarily used to approximate the subset posterior distribution $\hat{p}(\Theta \mid \mathcal{D}_k) = \sum_{j=1}^J \hat{z}_{k,j} p(\Theta \mid \mathcal{D}_k, \mathcal{M}_j)$ for each subset $k = 1, \dots, K$. By considering the second set of stacking weights $\hat{w} = \{\hat{w}_k\}_{k=1,\dots,K}$, the stacked full posterior distribution is $\hat{p}(\Theta \mid \mathcal{D}) = \sum_{k=1}^K \hat{w}_k \hat{p}(\Theta \mid \mathcal{D}_k) = \sum_{k=1}^K \hat{w}_k \sum_{j=1}^J \hat{z}_{k,j} p(\Theta \mid \mathcal{D}_k, \mathcal{M}_j)$, where $\Theta = \{\gamma, \Sigma\}$. This is a substantial simplification over meta-kriging (Guhaniyogi and Banerjee, 2018, 2019) that does not require any empirical approximation of posterior or predictive distributions. Again, the predictive random variable $Y_{\mathcal{U}}$ is recovered from (10) as

$$\hat{p}(Y_{\mathcal{U}} \mid \mathcal{D}) = \sum_{k=1}^K \hat{w}_k \sum_{j=1}^J \hat{z}_{k,j} p(Y_{\mathcal{U}} \mid \mathcal{D}_k, \mathcal{M}_j). \quad (11)$$

Each component $p(Y_{\mathcal{U}} \mid \mathcal{D}_k, \mathcal{M}_j)$ is a matrix-variate t distribution $T_{n',q}(\nu_k^*, \mu_{k,j}^*, V_{k,j}^*, \Psi_{k,j}^*)$. The mixture structure can inflate predictive covariances; disagreement tempering, described in Section 3.2, mitigates this effect. The latent spatial process is learned by sampling from $\hat{p}(\Omega_{\mathcal{U}} \mid \mathcal{D}) = \sum_{k=1}^K \hat{w}_k \hat{p}(\Omega_{\mathcal{U}} \mid \mathcal{D}_k)$, where $\hat{p}(\Omega_{\mathcal{U}} \mid \mathcal{D}_k) = \sum_{j=1}^J \hat{z}_{k,j} p(\Omega_{\mathcal{U}} \mid \mathcal{D}_k, \mathcal{M}_j)$.

3. Approximation Insights

We study the behavior of the DBPS posterior predictive distribution by characterizing and bounding its divergence from the true posterior predictive distribution using the Kullback-Leibler (KL) divergence. We then introduce disagreement tempering as a principled approach to mitigate predictive variance inflation and oversmoothing that commonly arise in distributed settings. Together, these results establish approximation guaranties for the

DBPS posterior predictive and show that, while distributed settings may exhibit predictive variance inflation, disagreement tempering mitigates this over-dispersion.

3.1 Kullback-Leibler Divergence from True Posterior Predictive

We investigate the behavior of DOUBLE BPS approximations to the posterior predictive distributions when the number of competitive models (J) and the number of partitions (K) grow in size. The reversed Kullback-Leibler divergence between the DOUBLE BPS posterior predictive in (11) and the true predictive distribution $p_t(\cdot | \mathcal{D})$ in (7), where $\{\alpha, \phi\}$ are as in the data-generating process, offers analytical tractability. Writing the true predictive distribution as $p_t(y | \mathcal{D}) = p(y | \mathcal{D}, \alpha_t, \phi_t) = \text{T}_{n,q}(y; \nu_t, M_t, V_t, \Psi_t)$ and the DOUBLE BPS approximation as $\hat{p}(y; \mathcal{D}) = \sum_{k=1}^K \hat{w}_k \sum_{j=1}^J \hat{z}_{k,j} \text{T}_{n,q}(y | \nu_{k,j}, M_{k,j}, V_{k,j}, \Psi_{k,j})$, and denoting $\hat{P}$ and P_t as the probability distributions corresponding to the DOUBLE BPS approximation and the true predictive probability distributions, respectively, the reverse KL divergence is

$$\begin{aligned} D_{\text{KL}}(\hat{P} \| P_t) &= \int_{y \in \mathbb{R}^{n \times q}} \sum_{k=1}^K \hat{w}_k \hat{p}_k(y; \mathcal{D}_k) \log \frac{\sum_{k=1}^K \hat{w}_k \hat{p}_k(y; \mathcal{D}_k)}{p_t(y | \mathcal{D})} dy \\ &= \sum_{k=1}^K \hat{w}_k \mathbb{E}_{\hat{p}_k} \left[\log \sum_{k=1}^K \hat{w}_k \hat{p}_k(y; \mathcal{D}_k) \right] - \sum_{k=1}^K \hat{w}_k \mathbb{E}_{\hat{p}_k} [\log p_t(y | \mathcal{D})] \\ &= \sum_{k=1}^K \hat{w}_k \mathbb{E}_{\hat{p}_k} \left[\log \frac{\sum_{k=1}^K \hat{w}_k \hat{p}_k(y; \mathcal{D}_k)}{p_t(y | \mathcal{D})} \right] \leq \sum_{k=1}^K \hat{w}_k \log \mathbb{E}_{\hat{p}_k} \left[\frac{\sum_{k=1}^K \hat{w}_k \hat{p}_k(y; \mathcal{D}_k)}{p_t(y | \mathcal{D})} \right], \end{aligned}$$

where $\hat{p}_k(y; \mathcal{D}_k) = \sum_{j=1}^J \hat{z}_{k,j} \text{T}_{n,q}(y | \nu_{k,j}, M_{k,j}, V_{k,j}, \Psi_{k,j})$ and the inequality follows from Jensen's inequality. Some further algebraic simplification yields

$$D_{\text{KL}}(\hat{P} \| P_t) \leq \log \prod_{k=1}^K \left\{ \sum_{k=1}^K \hat{w}_k \sum_{j=1}^J \hat{z}_{k,j} \mathbb{E}_{p_{k,j}} \left[\frac{\sum_{j=1}^J \hat{z}_{k,j} p(y | \mathcal{D}_k, \mathcal{M}_j)}{p_t(y | \mathcal{D})} \right] \right\}^{\hat{w}_k}. \quad (12)$$

The empirical behavior for the upper bound is studied using Monte Carlo experiments in Section C.2 of the Appendix.

3.2 Disagreement Tempering

Disagreement tempering (DT) aims to mitigate posterior predictive variability and predictive interval width when posterior predictive distributions are mixtures. A known issue with linear mixtures of predictive distributions is variance inflation due to a positive quantity called model disagreement. When component densities or forecasters differ in their means, the variance of the ensemble can be substantially larger than the variance of each model. Forecast combination theory shows that a linear pool tends to produce inflated credible intervals that may be too wide for practical purposes. In fact, the extra disagreement term in predictive variance can cause overdispersion, which has been studied in the literature on “linear pooling” (Knüppel and Krüger, 2022). In double Bayesian predictive stacking, the

Algorithm 3 Disagreement tempering of DBPS predictions

Input: $\hat{z} = \{\hat{z}_k = \{\hat{z}_{k,j}\} : k \in \{1, \dots, K\}, j \in \{1, \dots, J\}\}$: Stacking weights within subsets; $\hat{w} = \{\hat{w}_k : k = 1, \dots, K\}$: stacking weights between subsets; R : Number of predictive samples.

Output: $\{Y_{\mathcal{U}}^{DT(r)} : r = 1, \dots, R\}$: Disagreement-tempered predictive sample.

```

1: for  $r = 1, \dots, R$  do
2:   Sample  $k^{(r)} \sim \text{Multinom}(1, \hat{w})$ 
3:   Sample  $j^{(r)} \sim \text{Multinom}(1, \hat{z}_{k^{(r)}})$ 
4:   Sample  $Y_{\mathcal{U}}^{(r)} \sim p(Y_{\mathcal{U}} \mid \mathcal{D}_{k^{(r)}}, \mathcal{M}_{j^{(r)}})$ 
5:   Compute  $\mu_{k^{(r)}, j^{(r)}}^* = \mathbb{E}[Y_{\mathcal{U}} \mid \mathcal{D}_{k^{(r)}}, \mathcal{M}_{j^{(r)}}]$ 
6: end for
7: Compute centered global mean  $\mu_C^* = \frac{1}{R} \sum_{r=1}^R \mu_{k^{(r)}, j^{(r)}}^*$ 
8: for  $r = 1, \dots, R$  do
9:   Compute  $Y_{\mathcal{U}}^{DT(r)} = Y_{\mathcal{U}}^{(r)} - \mu_{k^{(r)}, j^{(r)}}^* + \mu_C^*$ 
10: end for
11: return  $\{Y_{\mathcal{U}}^{DT(r)} : r = 1, \dots, R\}$ 

```

predictive variance is

$$\text{Var}(\tilde{Y}_{\mathcal{U}}; \mathcal{D}) = \sum_{k=1}^K w_k \sum_{j=1}^J z_{k,j} \text{Var}(\tilde{Y}_{\mathcal{U}} \mid \mathcal{D}_k, \mathcal{M}_j) + \text{Dis}(\{m_{k,j}\}_{k,j}),$$

where $\tilde{Y}_{\mathcal{U}} = \text{vec}(Y_{\mathcal{U}})$, $m_{k,j} = \text{vec}(\mu_{k,j}^*) = \text{vec}(\mathbb{E}[Y_{\mathcal{U}} \mid \mathcal{D}_k, \mathcal{M}_j])$. The disagreement term $\text{Dis}(\{m_{k,j}\}_{k,j}) \succeq 0$ is derived in Section A.2. A simple solution to mitigate the effect of $\text{Dis}(\{m_{k,j}\}_{k,j})$ is centering, or “tempering”, the disagreement: subtract each model’s predictive mean so that all component distributions share a common center; form the mixture; and re-add a global mean if desired. In ensemble forecasting, this is known as the centered linear pool (Knüppel and Krüger, 2022). This procedure ensures that the disagreement term disappears, reducing the variability inflation and the width of the predictive intervals. Larger differences between means imply higher values of $\text{Dis}(\{m_{k,j}\}_{k,j})$ and therefore a greater impact of tempering disagreement (see Section A.2 for details).

In practice, we can apply a DT step without modifying the weights: we fit the stacking weights w_k as usual, but when making the final predictions, we first recenter each distribution (by subtracting the posterior predictive mean of the model) and then combine them; optionally, we add the ensemble mean (see Algorithm 3). This does not alter score-based weight optimization, but it can reduce the inflated uncertainty of the ensemble forecast. Hence, assimilation of Bayesian predictive distributions can suffer from extra variance due to model disagreement, and simple centering of predictions can mitigate this effect.

3.3 Alternative Approximate Methods

Our approach is based on an exact model with analytically accessible posterior distributions. Alternative approaches would include approximate methods. These can be broadly classified as methods that (i) approximate the true posterior distribution; and (ii) build scalable

Gaussian processes. Some distinctions are worth noting. Approximate posterior inference foregoes sampling from the exact posterior distribution (as is done in MCMC; see, e.g., Robert and Casella, 2004) and instead devises a faster algorithm to compute an approximate posterior (such as INLA, Rue et al., 2009; and variational Bayes, Blei et al., 2017). However, these approximate methods do not necessarily scale to massive datasets in terms of floating-point operations or storage complexity. Hence, their only possible benefit in the context of divide and conquer is to use them for each subset of the data to achieve faster convergence than MCMC. Scalable Gaussian processes, on the other hand, can be implemented with substantially lower storage and computational complexity and avoid the need for dividing and conquering massive data sets. While a comprehensive comparison with these alternative approaches is beyond the scope of a single article, we briefly discuss variational Bayes and nearest neighbor Gaussian processes in the context of our model.

Variational Bayes (see, e.g., Blei et al., 2017, for an excellent review from a statistical perspective) is prominent in machine learning for seeking an optimal approximation to intractable posteriors. Given that our posteriors are available in closed form, there is no apparent benefit to using variational Bayes or, for that matter, any iterative algorithm that needs to converge to an optimal approximation. Furthermore, the mean field variational approximation induces biased inference, which can be explicitly quantified for the MNIW model. We briefly discuss this in Section 3.3.1. Nearest neighbor Gaussian processes (introduced by Datta et al., 2016) scale statistical learning to massive datasets by building likelihood approximations (Vecchia, 1988) based on a sparse directed acyclic graphical model of the spatial topology. This results in a sparse precision matrix that enables fast computation. We briefly discuss this in Section 3.3.2.

3.3.1 VARIATIONAL INFERENCE

One advantage of the conjugate posterior distributions in our model is that we do not worry about biases resulting from approximate algorithms such as variational Bayes. It is well-known that mean-field variational approximations underestimate uncertainty in the posterior distribution (see, e.g., Section 3 in Ren et al., 2011, for an exposition using the Normal-Gamma Bayesian regression). Here, we offer a brief overview of this bias in a mean-field approximation of the posterior distribution for $\theta = \{\beta, \Sigma\}$ derived from (2).

Following Blei et al. (2017), we express our marginal distribution of Y as

$$\log p(Y) = \log \left(\frac{p(Y, \theta)}{p(\theta | Y)} \right) = \log p(Y, \theta) - \log p(\theta | Y) = \log \left(\frac{p(Y, \theta)}{q(\theta)} \right) + \log \left(\frac{q(\theta)}{p(\theta | Y)} \right) .$$

Let $q(\theta)$ be any valid probability density for θ . Multiplying both sides by $q(\theta)$ and integrating with respect to θ and noting that $\log p(Y) = \int q(\theta) \log p(Y) d\theta$, we obtain

$$\begin{aligned} \log p(Y) &= \int q(\theta) \log \left(\frac{p(Y, \theta)}{q(\theta)} \right) d\theta + \int q(\theta) \log \left(\frac{q(\theta)}{p(\theta | Y)} \right) d\theta \\ &= \text{ELBO}(q) + KL(q, p(\cdot | Y)) \geq \text{ELBO}(q) , \end{aligned}$$

where $\text{ELBO}(q) = \mathbb{E}[\log p(Y, \theta)] - \mathbb{E}[\log q(\theta)]$ is the *evidence lower bound* with $\mathbb{E}[\cdot]$ denoting expectation with respect to q , and $KL(q, p(\cdot | Y))$ is the Kullback-Leibler (KL) divergence between $q(\theta)$ to $p(\theta | Y)$. The divergence $KL(q, p(\cdot | Y))$ is always nonnegative, and $\log p(Y)$

is a constant for any given data Y , so the sum of $\text{ELBO}(q)$ and $KL(q, p)$ is a constant. Hence, $\text{ELBO}(q)$ acts as a lower bound for $\log p(Y)$. Given data Y , we can obtain the variational approximation of $p(\theta | Y)$ by either maximizing $\text{ELBO}(q)$ or minimizing $KL(q, p)$.

Focusing on maximizing $\text{ELBO}(q)$, the mean-field approximation seeks an optimal q within the class $q(\theta) = q_\beta(\beta)q_\Sigma(\Sigma)$. Therefore, to derive the optimal $q_\beta(\beta)$ we write

$$\begin{aligned} \text{ELBO}(q) &= \mathbb{E}_q[\log p(Y, \beta, \Sigma)] - \mathbb{E}_q[\log q(\beta) + \log(\Sigma)] \\ &= \mathbb{E}_{q_\beta} \{ \mathbb{E}_{q_\Sigma}[\log p(Y, \beta, \Sigma)] - \mathbb{E}_{q_\Sigma}[\log q(\beta) + \log q(\Sigma)] \} \\ &= \mathbb{E}_{q_\beta} \{ \mathbb{E}_{q_\Sigma}[\log p(Y, \beta, \Sigma)] - \log q(\beta) \} + \text{const} := \text{ELBO}(q_\beta), \end{aligned} \quad (13)$$

where “const” absorbs quantities free of β . Equation (13) reveals that $\text{ELBO}(q_\beta)$ is the negative Kullback-Leibler divergence between $q_\beta(\beta)$ and a density given by $\log q_\beta^*(\beta) = \mathbb{E}_\Sigma[\log p(Y, \beta, \Sigma)] + \text{constant}$. Therefore, we maximize $\text{ELBO}(q_\beta)$ when we set $q_\beta(\beta)$ as

$$q_\beta^*(\beta) \propto \exp \{ \mathbb{E}_{q_\Sigma}[\log p(Y, \beta, \Sigma)] \} \propto \exp \{ \mathbb{E}_{q_\Sigma}[\log p(\beta | Y, \Sigma)] \}, \quad (14)$$

where the last expression is a simple consequence of the fact that the full conditional distribution $p(\beta | \Sigma, Y)$ is proportional to the joint distribution $p(Y, \beta, \Sigma)$. Analogously,

$$q_\Sigma^*(\Sigma) \propto \exp \{ \mathbb{E}_{q_\beta}[\log p(Y, \beta, \Sigma)] \} \propto \exp \{ \mathbb{E}_{q_\beta}[\log p(\Sigma | Y, \beta)] \} \quad (15)$$

is the optimal mean-field approximation for the marginal posterior distribution of Σ . This draws a resemblance to the Gibbs sampler. While the Gibbs sampler draws samples from the full conditionals, the mean-field variational algorithm computes an optimal approximation based on the full conditional distribution.

The variational approximations in (14) and (15) are available explicitly. Focusing on the Bayesian updating described below (2) in Section 2.1 from the prior to a single dataset $\mathcal{D}_1 = \{Y, X\}$ with fixed row-covariance V , we easily see that $p(\beta | Y, \Sigma) = \text{MN}(\beta | Mm, M, \Sigma)$, where $M^{-1} = M_0^{-1} + X^T V^{-1} X$ and $m = m_0 + X^T V^{-1} Y$. Therefore,

$$\log q_\beta^*(\beta) = \text{const} - \frac{1}{2} \text{tr} \{ \zeta^{-1} (\beta - Mm) M^{-1} (\beta - Mm) \} = \log \text{MN}(\beta | Mm, M, \zeta), \quad (16)$$

where $\zeta^{-1} = \mathbb{E}_{q_\Sigma}[\Sigma^{-1}]$. Analogous calculations using (15) reveal that

$$\log q_\Sigma^*(\Sigma) = \text{const} - \frac{\nu_0 + q + p + n + 1}{2} \log \det(\Sigma) - \frac{1}{2} \text{tr} \left(\tilde{\Psi} \Sigma^{-1} \right) = \log \text{IW}(\Sigma | \tilde{\Psi}, \tilde{\nu}), \quad (17)$$

where $\tilde{\nu} = \nu_0 + p + n$ and $\tilde{\Psi} = \Psi_0 + \mathbb{E}_{q_\beta}[S_\beta(\beta) + S_Y(\beta)]$, with $S_\beta(\beta) = (\beta - M_0 m_0)^T M_0 (\beta - M_0 m_0)$ and $S_Y(\beta) = (Y - X\beta)^T V^{-1} (Y - X\beta)$ being the prior scatter and likelihood scatter, respectively. The expression for $\tilde{\Psi}$ simplifies, using elementary matrix algebra, to

$$\tilde{\Psi} = \Psi_0 + m_0^T M_0 m_0 + Y^T V^{-1} Y - m^T M m + \mathbb{E}_{q_\beta} [(\beta - Mm)^T M^{-1} (\beta - Mm)]. \quad (18)$$

The variational mean-field algorithm iterates between (16) and (17) as follows. Given the variational approximations $q_\beta^{(t)}(\beta)$ and $q_\Sigma^{(t)}(\Sigma)$ at iteration t , we compute

$$q_\beta^{(t+1)}(\beta) = \text{MN} \left(\beta | Mm, M, \zeta^{(t)} \right) \quad \text{and} \quad q_\Sigma^{(t+1)}(\Sigma) = \text{IW} \left(\Sigma | \tilde{\Psi}^{(t)}, \tilde{\nu} \right)$$

where $\zeta^{(t)} = \left(\mathbb{E}_{q_{\Sigma}^{(t)}}[\Sigma^{-1}] \right)^{-1}$ and $\tilde{\Psi}^{(t)} = \Psi_0 + m_0^T M_0 m_0 + Y^T V^{-1} Y - m^T M m + p \zeta^{(t)}$. The $\zeta^{(t)}$ appearing as the last term in the expression for $\tilde{\Psi}^{(t)}$ is obtained by simplifying the expectation in the last term of (18) with respect to $q_{\beta}^{(t+1)}(\beta)$. The sequence $\{\zeta^{(t)}\}$ is updated using recursion,

$$\zeta^{(t+1)} = \left(\mathbb{E}_{q_{\Sigma}^{(t+1)}}[\Sigma^{-1}] \right)^{-1} = \frac{\tilde{\Psi}^{(t)}}{\tilde{\nu}} = \frac{\Psi^* + p \zeta^{(t)}}{\tilde{\nu}}, \quad (19)$$

where $\Psi^* = \Psi_0 + m_0^T M_0 m_0 + Y^T V^{-1} Y - m^T M m$ does not depend on t .

Denoting $\lim_{t \rightarrow \infty} \zeta^{(t)} = \zeta$, we find ζ by taking the limits of both sides of (19) as $t \rightarrow \infty$. This yields $\zeta = \frac{\Psi^*}{\tilde{\nu} - p}$ and, therefore, $\lim_{t \rightarrow \infty} \tilde{\Psi}^{(t)} = \Psi^* + p \zeta = \frac{\tilde{\nu} \Psi^*}{\tilde{\nu} - p}$. While the exact marginal posterior distribution of Σ is $p(\Sigma | Y) = \text{IW}(\Sigma | \Psi^*, \nu)$ with $\nu = \nu_0 + n$, its variational approximation, upon convergence, yields $\text{IW}\left(\Sigma | \frac{\tilde{\nu} \Psi^*}{\tilde{\nu} - p}, \tilde{\nu}\right)$. The true marginal posterior distribution $p(\beta | Y)$ is a matrix-variate T distribution, while the variational approximation converges to the matrix-normal $\text{MN}\left(\beta | M m, M, \frac{\Psi^*}{\tilde{\nu} - p}\right)$. These distinctions between the true marginal posteriors and their variational approximations typically lead to underestimations of the variability. While these biases are mitigated for large sample sizes (under broad regularity conditions), applying VB to subsets of the data will induce discrepancies that are unnecessary since we have full analytical tractability for the joint posterior distributions.

3.3.2 SCALABLE GAUSSIAN PROCESS MODELS

Scalable Gaussian processes have been a prominent field of research in spatial statistics and are too vast to be comprehensively reviewed here (see, e.g., Banerjee, 2017; Heaton et al., 2017, and discussions therein). Rather than dividing and conquering massive datasets or building faster algorithms to diminish iteration complexity, this approach builds approximate Gaussian processes that ensure feasible computational and storage requirements. For subsequent comparisons with our approach, we briefly outline one particular class of models that builds a sparse approximation of the spatial precision matrix.

To see how sparsity accrues computational benefits, consider the model in (4) with the posterior distribution given in (6). Sampling from this posterior distribution requires the Cholesky decomposition of the $n \times n$ spatial covariance matrix $\rho_{\phi}(\mathcal{S}, \mathcal{S})$ or its inverse, which is computationally unfeasible for large n . Zhang et al. (2021) investigates computational strategies for the MNIW family using the nearest neighbor Gaussian process (NNGP, Datta et al., 2016; Zhang et al., 2019). The NNGP specifies a sparse spatial precision matrix $\tilde{\rho}_{\phi}(\mathcal{S}, \mathcal{S})^{-1}$ using the likelihood approximation developed by Vecchia (1988) and extends it to a well-defined spatial process over the entire domain. We provide a brief discussion of this construction and leave the details to Section 2.2.2 of Zhang et al. (2021).

The sparse precision matrix is $\tilde{\rho}_{\phi}(\mathcal{S}, \mathcal{S})^{-1} = H_{\rho}^T H_{\rho}$, where $H_{\rho} = D_{\rho}^{-1/2}(I - B_{\rho})$ is constructed using a directed acyclic graph based on a fixed topological order of the n spatial locations, say $(s_1, \dots, s_n)$. We construct neighbor sets $\mathcal{N}(s_i)$ for each s_i to comprise at most m nearest neighbors of s_i from the set of locations preceding it in the topological order. We construct B_{ρ} to be a sparse strictly lower-triangular matrix with the (i, j) -th entry 0 if $s_j \notin \mathcal{N}(s_i)$ for $j > i$. All entries in the first row of B_{ρ} are 0 because $\mathcal{N}(s_1)$ is empty, as are all diagonal entries since s_j is not included in $\mathcal{N}(s_j)$. This ensures that

there are at most m non-zero entries in B_ρ . Let $i_1 < i_2 < \dots < i_m$ be the column indices in the i -th row of B_ρ that have non-zero entries. These non-zero entries are obtained by solving $m \times m$ linear systems $\rho_\phi(\mathcal{N}(s_i), \mathcal{N}(s_i))b_i = \rho_\phi(\mathcal{N}(s_i), s_i)$ for each b_i , $i = 2, \dots, m$, where $\rho_\phi(\mathcal{N}(s_i), \mathcal{N}(s_i))$ is the $m \times m$ spatial correlation matrix of $\omega(s)$ constructed using a spatial correlation function $\rho_\phi(s, s')$ over the neighbor sets $\mathcal{N}(s_i)$ and $\rho_\phi(\mathcal{N}(s_i), s_i)$ is the $m \times 1$ vector of spatial correlations between $\omega(s_j)$ and $\omega(s_i)$ for each $s_j \in \mathcal{N}(s_i)$. The matrix $D_\rho^{-1/2}$ is diagonal with elements given by $(1 - b_i^\top \rho_\phi(\mathcal{N}(s_i), s_i))^{-1/2}$. This completes the specification for H_ρ .

The above construction exempts us from computing the Cholesky decomposition. We are essentially *modeling* the Cholesky decomposition rather than *computing* it. Furthermore, the $V_\gamma^{*-1} = \begin{bmatrix} \frac{\alpha}{1-\alpha} X^\top X + M_0^{-1} & \frac{\alpha}{1-\alpha} X^\top \\ \frac{\alpha}{1-\alpha} X & H_\rho^\top H_\rho + \frac{\alpha}{1-\alpha} \mathbb{I}_n \end{bmatrix}$ in (6) is easily factored as

$$V_\gamma^{*-1} = \underbrace{\begin{bmatrix} \sqrt{\frac{\alpha}{1-\alpha}} X^\top & M_0^{-1/2} & O \\ \sqrt{\frac{\alpha}{1-\alpha}} \mathbb{I}_n & O & H_\rho^\top \end{bmatrix}}_{H_\gamma^{*\top}} \underbrace{\begin{bmatrix} \sqrt{\frac{\alpha}{1-\alpha}} X & \sqrt{\frac{\alpha}{1-\alpha}} \mathbb{I}_n \\ M_0^{-1/2} & O \\ O & H_\rho \end{bmatrix}}_{H_\gamma^*} = H_\gamma^{*\top} H_\gamma^*.$$

Sampling from (6) is achieved by sampling $\Sigma \sim \text{IW}(\Psi^*, \nu^*)$, computing the Cholesky decomposition of the $q \times q$ matrix (cheaply) $\Sigma = R_\Sigma^\top R_\Sigma$, forming an $n \times q$ matrix Z with elements drawn from $Z_{ij} \stackrel{iid}{\sim} N(0, 1)$, solving the matrix equation $H_\gamma^* A = Z R_\Sigma$ for A , and setting $\gamma = \mu_\gamma^* + A$. Computational feasibility is ensured by avoiding the Cholesky decomposition of any $n \times n$ spatial correlation matrices. Zhang et al. (2021) develops and analyzes more efficient algorithms that reformulate sampling from (6) as a sparse least-squares problem solved using the LSMR algorithm of Fong and Saunders (2011).

Predictive inference follows from the marginal posterior predictive distribution in (7), but with $M_\mathcal{U}$ and $V_{\Omega_\mathcal{U}}$ derived from the NNGP model. We define the neighbor set $\mathcal{N}(u)$ of an arbitrary location $u \in \mathcal{U}$ as comprising the m closest neighbors of u from $\{s_1, \dots, s_n\}$. That is, any arbitrary location in $\mathcal{U}$ where predictions are sought is deemed to appear after s_n in the topological order of locations. Using this neighbor set, we construct the $n' \times n$ matrix $M_\mathcal{U}$ similar to B_ρ . Each row of $M_\mathcal{U}$ comprises at most m non-zero elements occupying column indices $i_1 < i_2 < \dots < i_m$ in row i . These are obtained by solving the $m \times m$ system $\rho_\phi(\mathcal{N}(u_i), \mathcal{N}(u_i))b_i = \rho_\phi(\mathcal{N}(u_i), u_i)$ for each b_i , $i = 1, \dots, n'$. The $n' \times n'$ covariance matrix $V_{\Omega_\mathcal{U}} = D_\mathcal{U}$ is diagonal with entries $d_i = 1 - b_i^\top \rho_\phi(\mathcal{N}(u_i), u_i)$. Posterior predictive sampling now proceeds exactly as described below (7).

4. Computer Programs and Resources

All our subsequent analyses are implemented in native R and c++ using the **spBPS** package. All programs required to reproduce the analysis are publicly accessible from the GitHub repository [lucapresicce/Bayesian-Transfer-Learning-for-GeoAI](https://github.com/lucapresicce/Bayesian-Transfer-Learning-for-GeoAI) that links the Rcpp-based **spBPS** package. The reported results are from a standard laptop running an Intel Core i7-8750H CPU with 5 cores for parallel computation and 16 GB of RAM.

We fit a linear model of coregionalization (LMC, Banerjee et al., 2025b) and multivariate seemingly unrelated Bayesian additive regression trees (multivariate BART, Esser et al.,

2025) using `spBayes` and `suBART` packages, respectively. We also compare with machine learning methods and AI systems using a scalable platform for parallelized supervised and unsupervised machine learning algorithms offered by `h2o` (Fryda et al., 2024). We specifically fit distributed random forest (DRF), gradient boosting (GBM), deep neural network (DNN), and a fully automatic machine learning algorithm (AUTOML). For parallel implementations of DBPS, we employ R packages `doParallel`, and `foreach` (Microsoft and Weston, 2022; Microsoft and Weston, 2022). We map the interpolated spatial surfaces using `MBA` (Finley et al., 2026), while sampling from the matrix-variate normal and t distributions is achieved using `mvnfast` (Fasiolo, 2014). Section B specifies computational considerations and sensitivity to the number of data shards, K , for spatial “BIG” data analysis.

We build a Bayesian transfer learning engine to conduct amortized Bayesian inference (Zammit-Mangion et al., 2025) using DBPS. We implement a residual neural network (ResNet) (He et al., 2015) using the R interfaces supplied by `tensorflow` (Allaire et al., 2024) and `keras` (Kalinowski et al., 2024) for native `Python`.

5. Simulation Experiments

We evaluate computational and inferential performance of DOUBLE BPS, while comparing against multiple alternative methodologies. We present a selection of simulation results for multivariate models here and refer the reader to Section C for further experiments.

5.1 Predictive Coverage Performance

We evaluate the predictive coverage performance and computational efficiency of our proposed framework using a synthetic spatial data set comprising 2,250: $n = 2,000$ locations used for training and $u = 250$ held for predictive evaluations. We also include a design matrix X with $p = 2$ comprising an intercept and a single predictor whose values were sampled independently from a uniform distribution on $[0, 1]$, and a univariate ($q = 1$) response Y . The spatial coordinates are sampled uniformly over the unit square $[0, 1]^2$ and the spatial correlation matrix $n \times n$ over these coordinates using $\rho_\phi(s_i, s_j) = \exp(-\phi \|s_i - s_j\|)$ with $\phi = 4$. The response is generated according to a Gaussian process model with regression coefficients $\beta = (1.0, 0.5)^\top$, spatial variance $\sigma^2 = 1$, and nugget variance $\tau^2 = 0.25$, which corresponds to $\alpha = 0.8$.

Although our methodology is inherently multivariate, we restricted this study to a univariate response and a moderate sample size. This design allows direct comparison with established gold-standard approaches: full Gaussian process models (Full GP) and nearest-neighbor Gaussian process (NNGP) models. Larger or multivariate datasets would render repeated full GP analyses infeasible, yet these settings are sufficient to evaluate predictive coverage, MSPE, and interval width across competing methods.

We assess distributed learning via DBPS, varying the number of subsets $K \in \{5, 10, 20\}$, with candidate models defined over a grid of hyperparameters $\alpha \in \{0.7, 0.8, 0.9\}$ and $\phi \in \{3, 4, 5\}$. We evaluate the impact of the disagreement term on the predictive interval width by applying the disagreement tempering technique described in Section 3.2 to double Bayesian predictive stacking predictions. For comparison, we fit NNGP models with neighbor numbers ranging from $m \in \{5, 10, 20\}$ and a full GP model as a comparison. For both simulation-based methods, we specify the same non-informative default priors and

Model	Time (sec)	Pred. Int. Width	MSPE	Emp. Coverage
FULL GP	1654 (1626, 1683)	2.00 (1.96, 2.05)	0.264 (0.222, 0.293)	0.945 (0.925, 0.969)
NNGP ($m = 20$)	120 (116, 130)	2.01 (1.97, 2.08)	0.264 (0.220, 0.290)	0.947 (0.923, 0.971)
NNGP ($m = 10$)	39 (32, 40)	2.01 (1.97, 2.07)	0.264 (0.221, 0.288)	0.949 (0.931, 0.969)
NNGP ($m = 5$)	17 (16, 18)	2.01 (1.98, 2.07)	0.264 (0.223, 0.291)	0.948 (0.929, 0.968)
DBPS ($K = 5$)	4.8 (4.7, 5)	2.03 (1.97, 2.10)	0.265 (0.222, 0.283)	0.946 (0.926, 0.966)
DBPS-DT ($K = 5$)	4.8 (4.7, 5)	1.96 (1.89, 2.03)	0.264 (0.222, 0.281)	0.939 (0.918, 0.959)
DBPS ($K = 10$)	2 (1.7, 2.1)	2.06 (1.96, 2.11)	0.268 (0.247, 0.305)	0.955 (0.938, 0.974)
DBPS-DT ($K = 10$)	2 (1.7, 2.1)	1.99 (1.91, 2.06)	0.267 (0.248, 0.304)	0.942 (0.913, 0.963)
DBPS ($K = 20$)	1.6 (1.5, 1.7)	2.19 (2.11, 2.27)	0.271 (0.243, 0.306)	0.960 (0.941, 0.978)
DBPS-DT ($K = 20$)	1.6 (1.5, 1.7)	2.04 (1.99, 2.14)	0.271 (0.242, 0.307)	0.947 (0.922, 0.965)

Table 1: Average predictive interval width, MSPE, empirical coverage at 95%, and computation time (in seconds) for different specifications of NNGP, DBPS, and full Gaussian process models. These quantities are computed for each of our 50 replicated datasets. We present their means and the 2.5% and 97.5% quantiles calculated over the 50 replicates. Boldface values indicate the best performance with respect to each metric.

2,000 MCMC samples. Each method is applied to $B = 50$ replicated datasets to obtain a frequentist evaluation for each method and assess variability in predictive performance.

Table 1 summarizes the results in terms of the average predictive interval width, the mean squared prediction error (MSPE), empirical coverage, and computational time in seconds for the complete Gaussian process (FULL GP), nearest neighbor Gaussian process (NNGP), and double predictive stacking models. To be precise, we present summaries of the distributions of these quantities over the $B = 50$ replicated datasets. Hence, we present the means and the 2.5th and 97.5th percentiles (in parentheses) of these quantities over the $B = 50$ replicated values. In Table 1, we denote the application of disagreement tempering to double predictive stacking, or not, using the acronyms DBPS-DT and DBPS, respectively.

From Table 1, we see that DBPS-based models tend to achieve predictive accuracy and coverage comparable to the NNGP and full GP models while dramatically reducing the computation time with respect to the latter. In particular, while most of the 95% intervals presented in Table 1 show large overlap and, therefore, not statistically significant differences, we note that disagreement tempering (DT) provides substantial improvements to DBPS. Indeed, DT introduces a net advantage in terms of smaller predictive interval widths while also retaining a plausible empirical coverage (just one percent lower than the nominal 95%). The computational time for the double predictive stacking decreases as the number of subsets increases, highlighting the scalability of the framework without compromising inferential quality. These findings demonstrate that a distributed approach can efficiently replicate the predictive performance of full Gaussian process models, making it a practical alternative for large spatial datasets. These results also prove that DOUBLE BPS achieves substantial computational savings relative to full GP and NNGP models while maintaining comparable predictive accuracy and coverage. This simulation experiment illustrates the scalability of the framework without compromising inference quality when compared with nearest-neighbor Gaussian process models.

Distributed approaches (e.g., Guhaniyogi and Banerjee, 2018) can lead to prediction oversmoothing, but this phenomenon is not manifested strongly in double BPS. However, they clearly inflate predictive variability as the number of subsets grows. Consequently, increasing the number of subsets induces wider predictive intervals and some extra coverage. The origin of this extra width of the predictive interval may arise from a disagreement term,

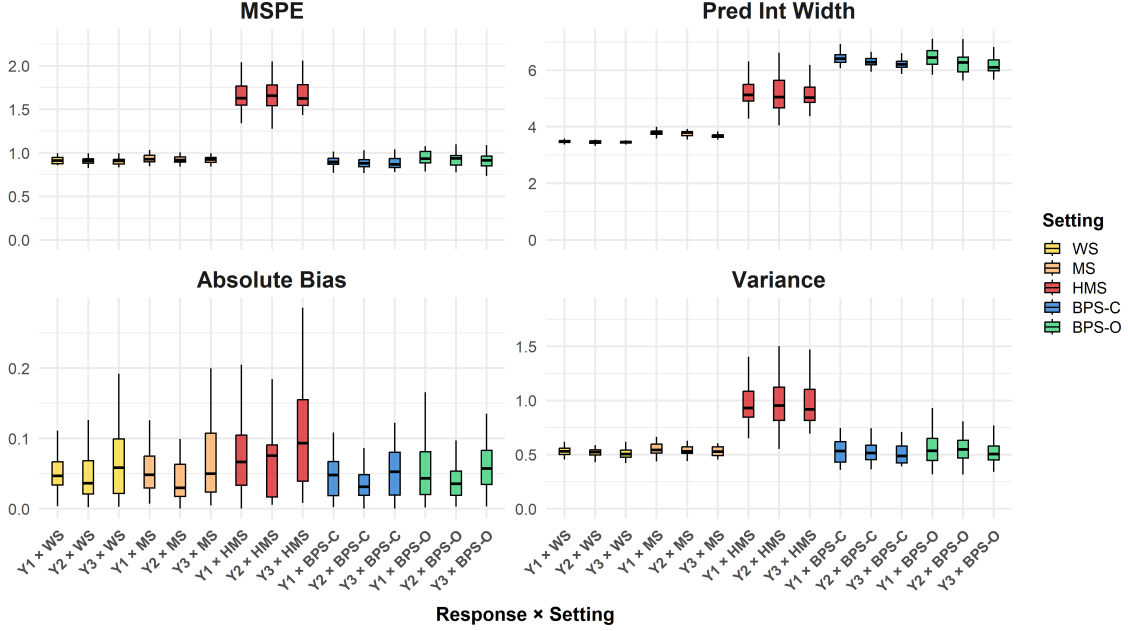


Figure 2: Predictive MSPE, interval width, absolute bias, and variance boxplot across responses and settings from 50 replications.

often associated with linear pooling forecasts (Knüppel and Krüger, 2022). However, the results in Table 1 confirm that the effect is negligible.

5.2 Transfer Learning in $\mathcal{M}$ -closed & $\mathcal{M}$ -open Settings

This simulation study serves a dual purpose: it assesses the performance of DBPS under $\mathcal{M}$ -closed and $\mathcal{M}$ -open settings and, crucially, it investigates the sensitivity of predictive inference to the choice of candidate values for $\{\alpha, \phi\}$. To this end, we consider both the $\mathcal{M}$ -closed scenario, in which the true spatial parameters lie in the candidate grid, and the $\mathcal{M}$ -open scenario, in which they do not. This directly addresses the practical question of what happens when the prior elicitation or automatic selection of the $\{\alpha, \phi\}$ grid is imperfect or coarse. While exploring how DOUBLE BPS behaves in the $\mathcal{M}$ -closed and $\mathcal{M}$ -open settings, we compare with the exact transfer learning framework that we devise in Section 2.1. In the latter, the model specification is characterized by different values of α and ϕ representing (i) a well-specified (WS) setting with the data generating values $\{\alpha = 0.8, \phi = 4\}$; (ii) a moderately misspecified (MS) setting with $\{\alpha = 0.45, \phi = 6.63\}$; and (iii) a highly misspecified (HMS) setting with $\{\alpha = 0.25, \phi = 50\}$.

Double BPS was tested on the $\mathcal{M}$ -closed and $\mathcal{M}$ -open settings. The former considers situations where the true model exists and is identified within a finite set of considered models. Here, the “true” model is the one such that $\{\alpha = 0.8, \phi = 4\}$. Then, for DOUBLE BPS under closed setting (BPS-C) we specify $J = 9$ competitive models with $\alpha \in \{0.75, 0.80, 0.85\}$ and $\phi \in \{2, 4, 6\}$ that yield effective spatial ranges in the percentage of maximum point inter-distance of 105%, 53%, 35% respectively, including the true model as one of the possible

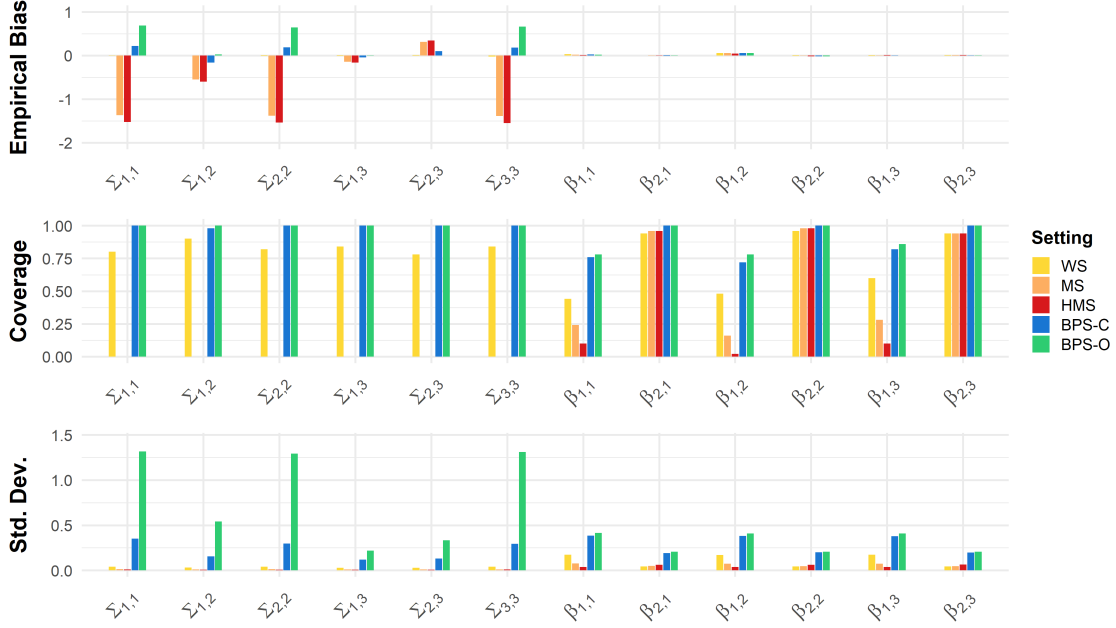


Figure 3: Average posterior bias, coverage, and standard deviation across parameters and settings from 50 replications.

candidates. Conversely, in the $\mathcal{M}$ -open setting, even though the true model exists, it cannot be fully specified. Thus, for DOUBLE BPS under open settings (BPS-O), we randomly define $J = 9$ candidate models. In particular, we uniformly sample 3 values for $\alpha \in (0, 1)$ and 3 values for $\phi \in (0, 50)$. We perform the experiment using 50 replications. Each replicate consists of values of the $n \times q$ outcome Y generated from (4) with $n = 5,000$, $q = 3$, and $p = 2$, the matrix X includes an intercept as its first column and a predictor generated from a standard uniform distribution over $[0, 1]$, $\beta = \begin{bmatrix} -0.75 & 1.05 & -0.35 \\ 2.20 & -1.10 & 0.45 \end{bmatrix}$ and $\Sigma = \begin{bmatrix} 2.00 & 0.80 & 0.20 \\ 0.80 & 2.00 & -0.45 \\ 0.20 & -0.45 & 2.00 \end{bmatrix}$. The $n \times n$ spatial correlation matrix V is specified using an exponential correlation function with $\phi = 4$ and $\alpha = 0.8$.

Figure 2 presents (i) mean square prediction error (MSPE); (ii) predictive interval width; (iii) absolute bias; and (iv) variance. We present boxplots for the distribution of each metric over the 50 replicates. This is made for each response and for each setting. In terms of predictive MSPE, absolute bias, and variance, the settings BPS-C and BPS-O exhibit slightly better performance. However, this seems to compromise the predictive interval width, as the uncertainty in the predictions is much higher. Approaches that estimate $\{\alpha, \phi\}$, rather than fix them, are somehow expected to introduce more uncertainty. In addition, we find no evidence of any difference between BPS-C and BPS-O for any metric. This is surprising, as it suggests the reliability of DBPS even in the transfer learning setting we devised in Section 2.3. Finally, irrespective of $\mathcal{M}$ -closed or $\mathcal{M}$ -open settings, it is more convenient to specify a set of candidate models using DOUBLE BPS instead of trying to fix $\{\alpha, \phi\}$.

Figure 3 presents posterior inference for (i) average empirical bias; (ii) average coverage; and (iii) average standard deviation, where the average is taken over the 50 replications. As expected, misspecification induces empirical bias in posterior estimates. The top panel

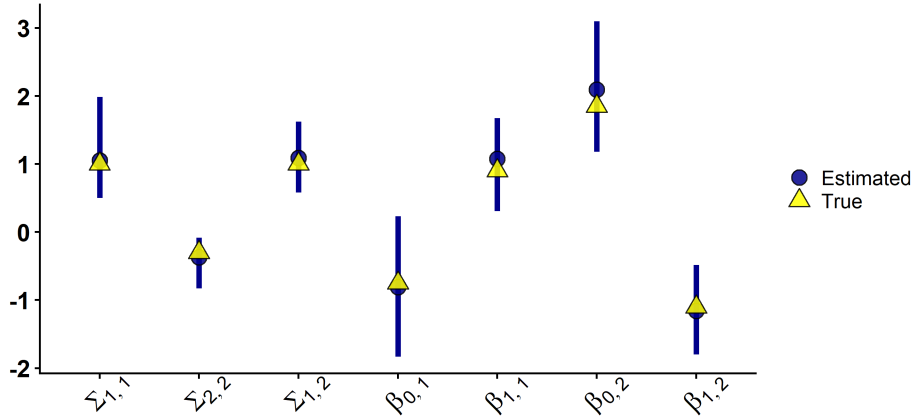


Figure 4: Amortized posterior credible intervals for parameters. True parameters in yellow.

reports how higher levels of bias are associated with models that are farther away from the truth. Here, DOUBLE BPS is placed in the middle for both $\mathcal{M}$ settings. The middle plot in Figure 3 shows that Σ is the parameter most affected by misspecifications. Its elements are well captured only by DOUBLE BPS, with coverage close to nominal, followed by WS specification, which performs worse. A similar pattern holds for β , where only DOUBLE BPS ensures adequate coverage. This reflects greater posterior variability in stacking approaches, as shown in the bottom panel reporting posterior standard deviations between settings. Interestingly, the inferential performance is weakest for spatial variance, which is not identifiable from the data obtained (Zhang, 2004). DOUBLE BPS behaves very similarly among $\mathcal{M}$ -closed and $\mathcal{M}$ -open. See Section C for additional simulation experiments.

Although DBPS outperforms exact transfer learning (Section 2) in the $\mathcal{M}$ -closed and $\mathcal{M}$ -open settings, the latter performs competitively with improved predictive performance in the WS and MS settings than in the HMS settings. This empirically demonstrates the robustness of DBPS with respect to the choice of the candidate grid for spatial parameters, confirming that stacking is resilient to hyperparameter misspecification, adapting to concentrate predictive mass on the closest approximating models. Our overall findings appear consistent with theoretical insights that Gaussian processes tend to deliver good predictive performance even for misspecified covariance functions in fixed domains (Stein, 1988; Stein and Handcock, 1989).

5.3 Amortized Bayesian Inference

We perform transfer learning by supervising a neural network using the output of DOUBLE BPS to deliver amortized Bayesian inference. We generate $N = 100$ instances of Y from (4) using a fixed realization of Ω for $q = 2$ correlated outcomes, $n = 500$ spatial locations that remain fixed across the datasets, and a fixed design matrix X with $p = 2$ comprising an intercept and a single predictor whose values were sampled independently from a uniform distribution over $[0, 1]$. The true regression coefficients are fixed at $\beta = \begin{bmatrix} -0.75 & 1.85 \\ 0.9 & -1.10 \end{bmatrix}$, with $\Sigma = \begin{bmatrix} 1 & -0.3 \\ -0.3 & 1 \end{bmatrix}$, $\alpha = 0.8$, and $\rho_\phi(s_i, s_j) = \exp(-\phi\|s_i - s_j\|)$ with $\phi = 4$.

We train the neural network in a supervised learning framework using $R = 250$ posterior samples. We apply DOUBLE BPS to each generated dataset with $K = 5$ subsets, $\alpha \in \{0.7, 0.8, 0.9\}$, and $\phi \in \{3, 4, 5\}$. These yield 100 instances of $\{Z, \Theta\}$, where $Z = [Y : X] \in \mathbb{R}^{n \times (q+p)}$, while $\Theta \in \mathbb{R}^{[(qp)+(q(q+1)/2)+(nq)] \times 3}$ comprises the posterior quantiles ($\{2.5\%, 50\%, 97.5\%\}$) for the distinct elements of $\{\beta, \Sigma, \Omega\}$.

We supervise a deep neural network comprising 3 hidden layers with 128, 256, and 512 nodes, with ReLU activations. The residual network is trained over 50 epochs (with 24 batches per epoch) by minimizing $\hat{\psi} = \arg \min_{\psi} 1/N \sum_{i=1}^N L(g_{\psi}(Z_i), \Theta_i)$, where $g_{\psi}(Z_i)$ is the neural output from Z_i trained by the Θ_i obtained using DOUBLE BPS for each of the $N = 100$ generated data sets. This training is implemented in *Keras* using the Adam optimizer with mean squared error (MSE) loss $L(\cdot, \cdot)$. For evaluation, we apply the trained model to unseen datasets with the same dimensions.

Figure 4 displays the amortized posterior credible intervals (blue bars) for $\{\beta, \Sigma\}$, along with the true values of the parameters (yellow triangles). The amortized 95% credible intervals capture all true values and the medians (50%) align closely with the true values, highlighting the effectiveness of the deep network in recovering posterior summaries. Figure 5 displays amortized inference for Ω . We compare the results from amortized inference with the true values of Ω , and the DOUBLE BPS prediction for the 50th quantile presented in the first and second columns of Figure 5, respectively.

These results illustrate the strengths of amortized inference and transfer learning. Once trained, the deep network provides instantaneous posterior quantile estimates for new datasets, without requiring us to rerun DOUBLE BPS. This amortizes the computational cost for future tasks. Additionally, the model generalizes across a range of data-generating conditions, effectively enabling posterior transfer learning to new but structurally similar problems. This makes the approach especially useful in large-scale or resource-constrained applications where repeated full Bayesian inference is prohibitively expensive.

6. Data Analysis

6.1 Vegetation Index Data

Statisticians and machine learners face growing demands to analyze and study global warming datasets (see, e.g., Fisher, 1958; Nicholls, 1989; Friehe et al., 1991; O’Carroll et al., 2019). The sheer volume of such datasets and the process-driven models required for their analysis have naturally raised the question of migrating such analysis to AI platforms. Our current application analyzes vegetation index data from the Moderate Resolution Imaging Spectroradiometer (MODIS) developed by NASA. We specifically focus on “MOD13C1.061 - Terra Vegetation Indices 16-Day L3 Global 0.05 Deg Climate Modeling Grid” (Didan, 2021), which provides vegetation indices per pixel on a 0.05-degree climate modeling grid (3600 rows by 7200 columns of 5600-meter pixels). It contains 16-day global composites, cloud-free, with additional reflectance and angular information.

Modeling the Normalized Difference Vegetation Index (NDVI) and Red Reflectance (RR) jointly is scientifically important, as it separates and defines the specific biophysical factors influencing vegetation. NDVI is built on the fundamental principle that healthy vegetation absorbs red light and reflects near-infrared (NIR) light. By examining both the calculated NDVI and its red reflectance component, scientists can gain deeper insights into the struc-

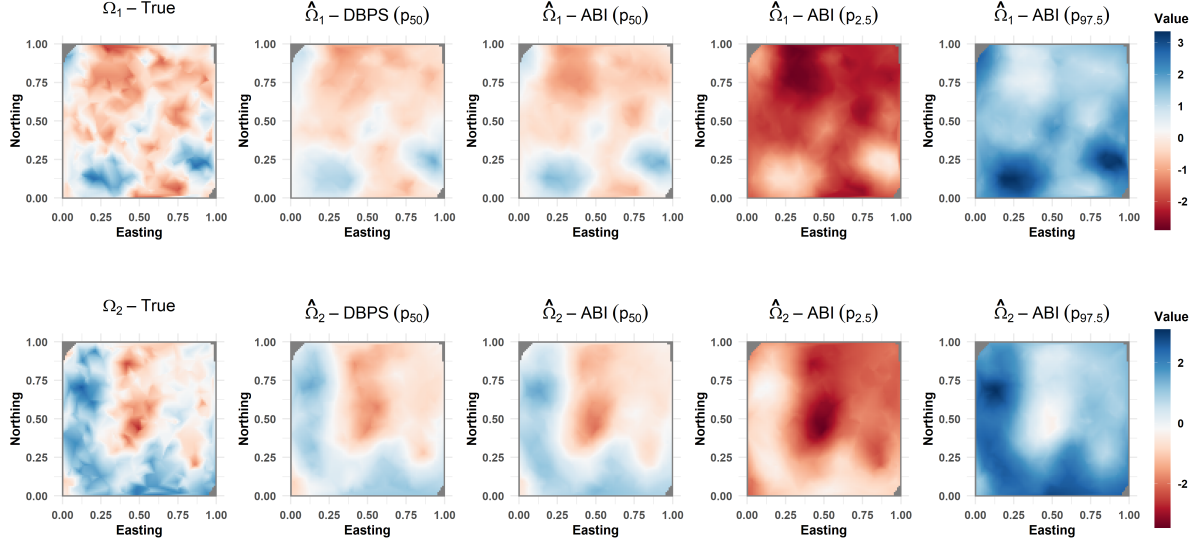


Figure 5: Surface interpolations for true spatial process, BPS prediction (50-th quantile), and amortized prediction of $\{50, 2.5, 97.5\}$ quantiles. Each row corresponds to an outcome.

Vegetation index	Mean	Std.Dev	Min.	Max.	Histogram	Boxplot
NDVI	8.593	0.517	6.909	9.469		
Red Reflectance	8.563	0.447	8.007	9.472		

Table 2: Summary statistics and visual representation of response variables.

tural and biochemical properties of vegetation. The solar zenith angle (SZA), which is the angle between the Sun and the point directly overhead, is a shared predictor that influences how much solar irradiation reaches the surface and is, therefore, crucial when assessing biomass and vegetation indices. NDVI and RR reflect vegetation’s capacity to absorb photo-synthetically active radiation, helping scientists understand the underlying mechanisms of climate change (Tucker, 1979; Sellers, 1985; Justice et al., 1985; Haque et al., 2024).

The data set comprises 1,500,000 locations. We randomly selected $n = 1,000,000$ sites for model fitting. To assess predictive performance and its stability with respect to the choice of evaluation locations, we constructed 5 disjoint held-out sets by randomly sampling $u = 250,000$ locations from the remaining sites and partitioning them into 5 non-overlapping subsets of 50,000 test locations each. All competing methods were fitted once on the full training set of n observations; predictive evaluation was then carried out independently on each of the 5 held-out sets. Reported metrics in Table 4 are averages over the 5 replicates. This protocol extends standard practice in large-scale spatial data benchmarking, where a single held-out set is typically reserved for predictive evaluation (see, e.g., Zhang et al., 2019). Rather than relying on a single split, we evaluate all competing methods on multiple independent, disjoint held-out sets, allowing us to assess not only predictive accuracy but

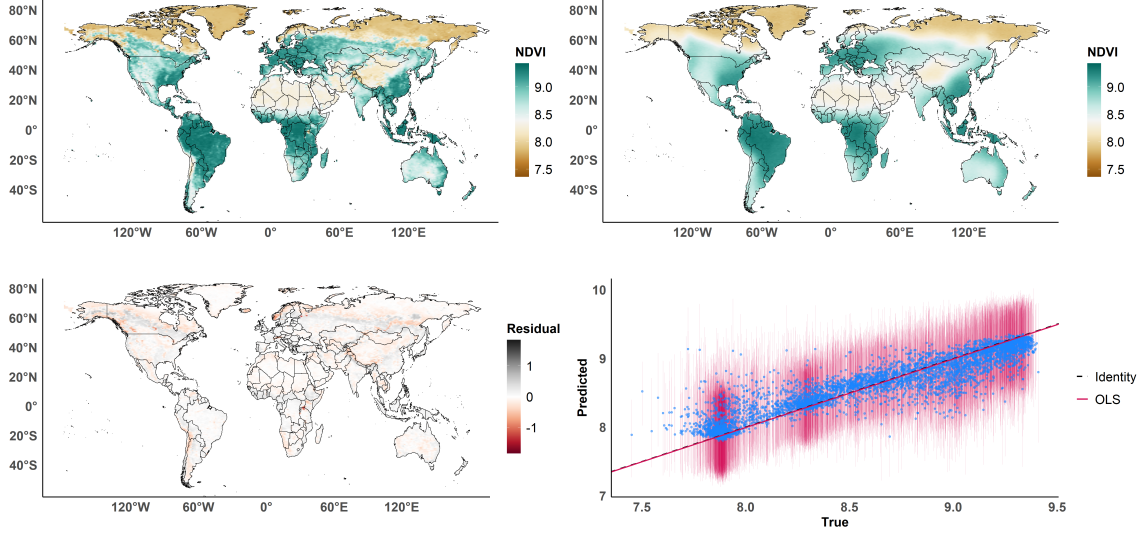


Figure 6: Left to right: Maps for test data (top left), predicted data (top right), and residual surface (bottom left) for NDVI. Predicted values scatterplot against the truth, with error bars for held-out values, is in the bottom right, alongside identity and OLS lines. Results correspond to $K = 2,000$.

also its stability. This yields a comprehensive out-of-sample assessment over a total of 250,000 held-out multivariate observations while keeping fitting fixed, as repeated refitting and predictions at this scale would be prohibitive for competing methods. Following Zhang and Banerjee (2022), outcomes were logarithmically transformed and we labeled $\log(\text{NDVI} + 1)$ as NDVI and $\log(\text{RR} + 1)$ as RR. All variables, including SZA, were averaged over a 16-day window in May 2024. Table 2 summarizes the spatial response distribution. The maximum distance between sites is approximately 42,909 kilometers.

Our central scientific objective is to jointly predict NDVI and RR from massive, globally distributed datasets of this scale using GEOAI. We assess whether multivariate statistical models can deliver accurate, scalable, and timely predictions across millions of spatial locations while accounting for dependence structures in the solar zenith angle. These capabilities are crucial for their application in ecology, agriculture, and climate policy.

6.2 Results Using DBPS

To analyze Vegetation Index (VI) data, we conduct machine-generated EDA in Appendix D. The predictors comprise an intercept and the solar zenith angle for that location ($p = 2$). Based on the variogram analysis in Appendix D, we set $\alpha \in \{0.825, 0.909\}$, and $\phi \in \{0.049, 0.067\}$ respectively. We specify $\{\gamma, \Sigma\}$ in (5) using $m_0 = 0_{p \times q}$, $M_0 = 10\mathbb{I}_p$, $\Psi_0 = \mathbb{I}_q$, $\nu_0 = 3$, and finally opt for an exponential spatial covariance function. Following insights from Section C.3, we fix the subset size at $n_k \in \{250, 500\}$, leading to a number of subsets $K \in \{4,000, 2,000\}$ respectively. We use a random scheme to form the partition of the data set and present the results when $K = 2,000$ and in Table 3 for $K = 4,000$.

We fit the multivariate model developed in Section 2.3 with NDVI and RR comprising the $q = 2$ columns of Y . Figures 6 and 7 illustrate maps and predictive diagnostic corresponding

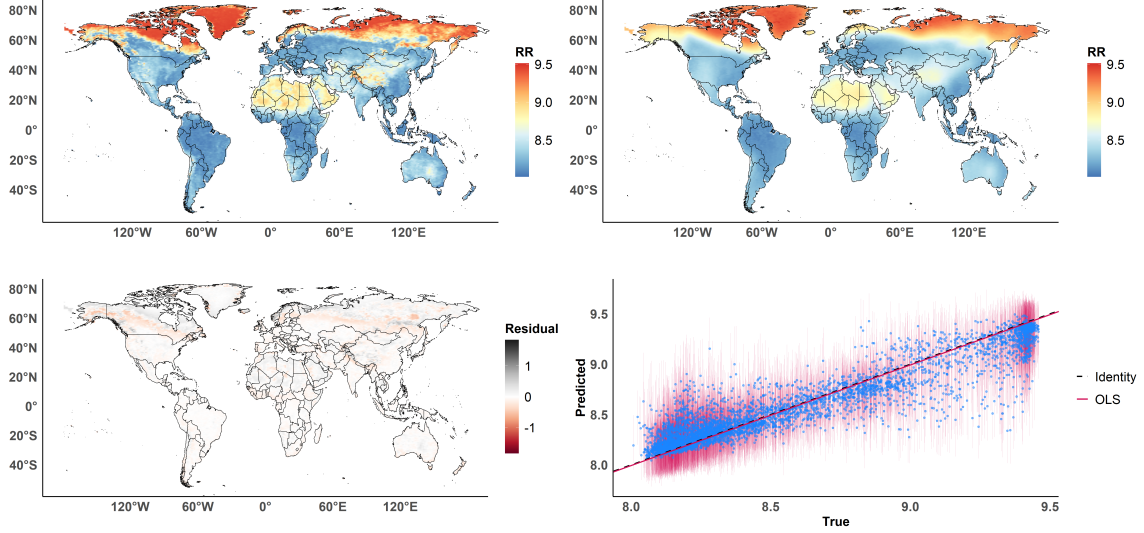


Figure 7: Left to right: Maps for test data (top left), predicted data (top right), and residual surface (bottom left) for RR. Predicted values scatterplot against the truth, with error bars for held-out values, is in the bottom right, alongside identity and OLS lines. Results correspond to $K = 2,000$.

to NDVI and RR, respectively, using $K = 2,000$ subsets. The top left panel in each figure presents the spatially interpolated map of the held-out test observations for the respective responses, revealing pronounced spatial variation: darker shades of green in NDVI represent higher values of detected biomass, while lighter shades of brown represent low biomass. In contrast, for RR, warmer colors in the red spectrum represent higher reflectance.

The top right panel of figures 6 and 7 shows the DBPS interpolated posterior mean surfaces, which are nearly indistinguishable from the observed test surfaces (top left). This suggests that automated DBPS effectively, perhaps even strikingly, recovers spatial patterns despite simplifications of the modeling over more elaborate statistical models (Banerjee, 2020; Zhang et al., 2025). The bottom left panel of each figure shows the map of pointwise prediction residuals at the held-out locations. The absence of systematic spatial structure in the residuals confirms that DBPS does not exhibit localized bias and adequately captures the global spatial dependence in both responses. The bottom right panel presents scatterplots of observed versus posterior mean predicted values at the held-out observations, with the identity line and the OLS regression line overlaid. The perfect overlap between the two lines confirms the absence of systematic predictive bias, and the tight concentration around them corroborates the quantitative accuracy reported in Table 4 for NDVI and RR, respectively.

We attempted comparisons with other Bayesian models, which represent the benchmark for spatial data analysis: the LMC, NNGP, and BART models. We were unable to fit any of them into our multivariate data because they exceeded memory. However, since NNGP is the gold standard in large-scale analysis, the restricted computational resources we use to fit the data set do not allow us to allocate the output. This highlights a main advantage of using DBPS when working on limited computational frameworks.

Parameter	BLM	DBPS ($K = 4,000$)	DBPS ($K = 2,000$)
$\beta_{0,\text{NDVI}}$	34.495 (34.392, 34.601)	1.670 (-0.570, 4.159)	1.814 (-0.787, 4.312)
$\beta_{1,\text{NDVI}}$	-2.708 (-2.719, -2.697)	0.716 (0.453, 0.955)	0.700 (0.431, 0.979)
$\beta_{0,\text{RR}}$	-16.924 (-17.014, -16.837)	-0.586 (-2.512, 1.498)	-0.633 (-2.408, 1.407)
$\beta_{1,\text{RR}}$	2.664 (2.655, 2.674)	0.962 (0.739, 1.162)	0.966 (0.749, 1.156)
Σ_{NDVI}	0.221 (0.220, 0.221)	0.182 (0.133, 0.245)	0.147 (0.114, 0.211)
$\Sigma_{\text{NDVI},\text{RR}}$	-0.167 (-0.168, -0.166)	-0.116 (-0.167, -0.081)	-0.093 (-0.132, -0.072)
Σ_{RR}	0.155 (0.154, 0.155)	0.118 (0.082, 0.165)	0.093 (0.073, 0.134)

Table 3: Vegetation Index data analysis parameter estimates for Bayesian conjugate linear model (BLM) and DBPS models. Parameter posterior summary 50 (2.5, 97.5) percentiles.

Table 3 compares the posterior estimates of the model parameters for DBPS with $K \in \{4,000, 2,000\}$, and the Bayesian conjugate linear model (BLM), which does not take into account spatial variability. The notably higher magnitudes of the intercepts in the non-spatial linear model are unsurprising, as the spatial random effects absorb much of the variation in the intercepts. The solar zenith angle is positively associated with RR and negatively associated with NDVI in the non-spatial model. This is also expected since higher levels of solar irradiation are associated with higher red reflectance and with more arid regions with less vegetation. However, the spatial models reveal that the slope for NDVI is significantly positive with solar irradiation after the spatial effects have absorbed previously unaccounted for latent or lurking factors that might have contributed to the negative slopes in the non-spatial model.

In addition to DOUBLE BPS (DBPS), LMC, NNGP, and Bayesian multivariate linear regression, we expanded the analysis by including four competitive algorithms: distributed random forest (DRF), gradient boosting (GBM), deep neural network (DNN), and a fully automatic machine learning algorithm (AUTOML). As detailed in Section B, we implement and perform the analysis using the `h2o` R package (Fryda et al., 2024). As no method for multivariate models is available within the `h2o` framework, we fit these algorithms separately for NDVI, and RR, considering the same explanatory variables. Then, we compute the empirical correlation among the predictions. Table 4 compares the different methods with regard to computational costs, predictive performances, and empirical correlation, averaged over the 5 disjoint held-out sets. These metrics are computed for each held-out set and summarized in terms of their means and standard deviations in the row below each model.

Predictive accuracy is assessed through three complementary criteria. The root mean square prediction error (RMSPE), reported separately for NDVI, RR, and their average, measures pointwise accuracy of the posterior mean predictions. The continuous ranked probability score (CRPS) is a strictly proper scoring rule (Gneiting and Raftery, 2007) that evaluates the full predictive distribution against each scalar observation, rewarding both sharpness and calibration; lower values indicate more accurate probabilistic predictions. The energy score (ES) extends this evaluation to the multivariate setting, assessing the joint predictive distribution over both responses simultaneously; particularly informative here with multivariate outcomes. The DBPS dominates both running time and predictive performances for all three metrics. With $K = 2,000$, it achieves RMSPE of (0.208, 0.163), CRPS of (0.116, 0.080), and ES of 0.150, substantially outperforming all competitors. Extremely close performance was achieved with $K = 4,000$, still dominating other models.

Model	Time (min)	RMSPE	CRPS	ES	$\varrho_{\text{NDVI,RR}}$
DBPS ($K = 2,000$)	15	[0.208, 0.163, 0.185]	[0.116, 0.080]	0.150	-0.905 (-0.918, -0.885)
SD	—	[0.0005, 0.0004, 0.0003]	[0.0003, 0.0001]	0.0002	0.002 (0.001, 0.002)
DBPS ($K = 4,000$)	4.5	[0.233, 0.184, 0.209]	[0.133, 0.094]	0.173	-0.900 (-0.920, -0.863)
SD	—	[0.0005, 0.0010, 0.0007]	[0.0001, 0.0004]	0.0003	0.003 (0.002, 0.005)
BLM	3	[0.471, 0.394, 0.432]	[0.276, 0.229]	0.370	-0.921 (-0.927, -0.916)
SD	—	[0.0010, 0.0009, 0.0009]	[0.0007, 0.0006]	0.0008	0.0001 (0.0001, 0.0001)
LMC	—	[-, -, -]	[-, -]	—	— (-, -)
SD	—	[-, -, -]	[-, -]	—	— (-, -)
NNGP ($m = 5$)	—	[-, -, -]	[-, -]	—	— (-, -)
SD	—	[-, -, -]	[-, -]	—	— (-, -)
BART	—	[-, -, -]	[-, -]	—	— (-, -)
SD	—	[-, -, -]	[-, -]	—	— (-, -)
GBM	3	[0.420, 0.346, 0.383]	[0.353, 0.275]	0.463	— (-, -)
SD	—	[0.0015, 0.0012, 0.0014]	[0.0011, 0.0009]	0.0013	— (-, -)
DRF	10	[0.421, 0.347, 0.384]	[0.353, 0.275]	0.463	— (-, -)
SD	—	[0.0016, 0.0013, 0.0014]	[0.0011, 0.0010]	0.0014	— (-, -)
DNN	26	[0.420, 0.347, 0.383]	[0.354, 0.274]	0.463	— (-, -)
SD	—	[0.0017, 0.0013, 0.0014]	[0.0012, 0.0009]	0.0014	— (-, -)
AUTOML	44	[0.420, 0.346, 0.383]	[0.353, 0.275]	0.463	— (-, -)
SD	—	[0.0015, 0.0013, 0.0014]	[0.0011, 0.0009]	0.0013	— (-, -)

Table 4: Vegetation Index data analysis computing time in minutes, RMSPE, CRPS, ES, and empirical correlation (ϱ) for candidate models. Root mean square prediction error(s) presentation [NDVI, red reflectance, average], continuous ranked probability score (s) presentation [NDVI, red reflectance]. Metrics are averaged over 5 different holdout sets and the row below each model reports their standard deviation across the 5 holdout sets.

The gap is particularly pronounced relative to machine learning methods, which achieve comparable values among themselves, but consistently yield RMSPE, CRPS, and ES values roughly two to three times larger than DBPS. The Bayesian conjugate multivariate linear model (BLM), while offering a proper predictive distribution, shows significantly lower (by almost 100%) pointwise accuracy, confirming the importance of spatial dependence modeling. For DBPS and BLM, Table 4 also reports posterior estimates of ϱ_{NDVI} and ϱ_{RR} . The results reveal a well-documented negative association between the two responses. Indeed, the spatial patterns in these indices are almost the reverse of each other as revealed in Figures 6 and 7. The conjugate Bayesian linear model estimates a higher negative correlation, while DBPS tends to underestimate it, considerably so as n_k decreases.

Recalling that all the analyses are produced on a personal laptop (with just 5 physical cores) with minimal human intervention, the total run time of only 4.5 minutes with $K = 4,000$ and 15 minutes with $K = 2,000$ for DBPS is impressive and confirms the quadratic dependence of the partition size discussed in Section C.3. Moreover, the strong dependence on the number of J competitive models is worth noting. With $J = 4$, this suggests a marginal computational burden for each competitive model.

7. Discussion

This manuscript devises a statistical modeling component for an artificially intelligent geospatial system (GEOAI) geared to analyze massive data with minimal human interven-

tion. The contribution harnesses analytically accessible multivariate statistical distributions in conjunction with Bayesian predictive stacking to deliver rapid inference by circumventing iterative algorithms that often require extensive human tuning. Our proposed GeoSpatial AI system relies on Bayesian transfer learning using double Bayesian predictive stacking to process massive amounts of streaming spatial data on high-performance CPU architectures.

Some additional remarks are warranted. The development here has been elucidated with a hierarchical matrix-variate spatial process framework. Although modeling simplifications have been introduced to minimize human intervention, we emphasize that DBPS seamlessly applies to more versatile but analytically intractable models. For spatially misaligned multivariate data, where not all variables have been observed in the same set of locations, we encounter missing entries in Y using analytically tractable closed-form distribution theory (Zhang and Banerjee, 2022). Should users wish to explore more complex models with unknown $\{\alpha, \phi\}$ or work with alternate multivariate models, we can implement Bayesian inference (using MCMC, Variational Bayes, or Laplace approximations) on each subset and invoke BPS to pool across the subsets of the data. The time for analyzing each subset increases due to model complexity, but scaling inference to massive datasets is still achieved by BPS—and this is achieved with considerable efficiency compared to divide-and-conquer approaches such as the geometric median, or GM, of posteriors (see meta-kriging in Guhaniyogi and Banerjee, 2018).

Future research can build upon recent work by Cabel et al. (2025) to enrich spatial-temporal dependence structures within DBPS and further accelerate amortized inference. Section 2.3 devises posterior inference as a “mixture of mixtures”. Following Yao et al. (2018), we prefer stacking to fitting a mixture model because the former is numerically robust and requires almost no human adjustment. Moreover, the structure of “mixture of mixtures” conceptually resembles the “Mixture of Experts” (MOE) that are adopted by AI platforms such as GPT-4 and Mistral. Disseminating our proposed methodology with our accompanying software, which is currently being migrated to R, is expected to significantly boost GEOSPATIAL AI systems. Future directions will also explore perceived potential of DBPS as a feeder for emerging amortized inference methods (Ganguly et al., 2023; Zammit-Mangion et al., 2025; Sainsbury-Dale et al., 2024) to achieve Bayesian inference. Rapid delivery of posterior estimates of the entire spatial process from DBPS will amount to more training data for amortized neural learners that can result in accelerated tuning for subsequent Bayesian inference. We do not see our proposed approach as a competitor to, but rather as supplementary to, amortized neural inference. See Section 5.3 for an effective representation. Such developments will be pursued as future research. We also seek to expand and fully investigate automated DOUBLE BPS using Markovian graphical structures across data subsets to further expedite and improve GEOSPATIAL AI systems.

Acknowledgments

Sudipto Banerjee acknowledges funding from the National Science Foundation through the Division of Mathematical Sciences (DMS grants 2113778 and 2515898); and funding from the National Institute of Health through the National Institute of General Medical Sciences (NIGMS grant R01GM148761).

Appendix A. Theoretical Derivations

A.1 Posterior and Predictive Matrix-Variate T Distributions

The joint posterior predictive for $Y_{\mathcal{U}}$ and the unobserved latent process $\Omega_{\mathcal{U}}$, can be recast by integrating out $\{\gamma, \Sigma\}$ from the joint conditional posterior predictive, that is

$$p(Y_{\mathcal{U}}, \Omega_{\mathcal{U}} | \mathcal{D}) = \int \text{MN}_{n',q}(Y_{\mathcal{U}} | X_{\mathcal{U}}\beta + \Omega_{\mathcal{U}}, (\alpha^{-1} - 1)\mathbb{I}_{n'}, \Sigma) \times \text{MN}_{n',q}(\Omega_{\mathcal{U}} | M_{\mathcal{U}}\Omega, V_{\Omega_{\mathcal{U}}}, \Sigma) \\ \times \text{MNIW}(\gamma, \Sigma | \mu_{\gamma}^*, V_{\gamma}^*, \Psi^*, \nu^*) d\gamma d\Sigma,$$

where $M_{\mathcal{U}} = \rho_{\phi}(\mathcal{U}, \mathcal{S})\rho_{\phi}^{-1}(\mathcal{S}, \mathcal{S})$ and $V_{\Omega_{\mathcal{U}}} = \rho_{\phi}(\mathcal{U}, \mathcal{U}) - \rho_{\phi}(\mathcal{U}, \mathcal{S})\rho_{\phi}^{-1}(\mathcal{S}, \mathcal{S})\rho_{\phi}(\mathcal{S}, \mathcal{U})$. We derive $p(\Omega_{\mathcal{U}}, Y_{\mathcal{U}} | \Sigma, \mathcal{D})$ by avoiding direct integration with respect to γ using the following augmented linear system

$$\underbrace{\begin{bmatrix} \Omega_{\mathcal{U}} \\ Y_{\mathcal{U}} \end{bmatrix}}_{\Upsilon} = \underbrace{\begin{bmatrix} 0_{n' \times p} & M_{\mathcal{U}} \\ X_{\mathcal{U}} & M_{\mathcal{U}} \end{bmatrix}}_M \underbrace{\begin{bmatrix} \beta \\ \Omega \end{bmatrix}}_{\gamma} + \underbrace{\begin{bmatrix} E_{\Omega_{\mathcal{U}}} \\ E_{Y_{\mathcal{U}}} \end{bmatrix}}_E, \quad E \sim \text{MN}_{2n',q}(0_{2n' \times q}, V_E, \Sigma),$$

where $V_E = \begin{bmatrix} V_{\Omega_{\mathcal{U}}} & V_{\Omega_{\mathcal{U}}} \\ V_{\Omega_{\mathcal{U}}} & (\alpha^{-1} - 1)\mathbb{I}_{n'} + V_{\Omega_{\mathcal{U}}} \end{bmatrix}$. We write the posterior distribution $p(\gamma | \mathcal{D})$ in Equation (6) as a linear equation, $\gamma = \mu_{\gamma}^* + E_{\gamma}$, with $E_{\gamma} \sim \text{MN}_{p+n,q}(O, V_{\gamma}^*, \Sigma)$, where E and E_{γ} are independent of each other. Then,

$$\Upsilon = M\mu_{\gamma}^* + ME_{\gamma} + E \sim \text{MN}_{2n',q}(M\mu_{\gamma}^*, V^*, \Sigma), \quad (20)$$

where $V^* = MV_{\gamma}^*M^{\top} + V_E$. This yields $p(\Upsilon | \Sigma, \mathcal{D}) = p(\Omega_{\mathcal{U}}, Y_{\mathcal{U}} | \Sigma, \mathcal{D})$ as the closed-form joint predictive distribution by integrating out Σ from $p(\Omega_{\mathcal{U}}, Y_{\mathcal{U}} | \Sigma, \mathcal{D})p(\Sigma | \mathcal{D})$ to get

$$\int \text{MNIW}(\Omega_{\mathcal{U}}, Y_{\mathcal{U}}, \Sigma | M\mu_{\gamma}^*, V^*, \Psi^*, \nu^*) d\Sigma = \text{T}_{2n',q}(\nu^*, M\mu_{\gamma}^*, V^*, \Psi^*)$$

which is a matrix-variate t random variable. Defining $\Upsilon = [\Omega_{\mathcal{U}}^{\top}, Y_{\mathcal{U}}^{\top}]^{\top}$ as a matrix of dimension $m \times q$, where $m = 2n'$, the predictive distribution is

$$p(\Upsilon | Y) = \int P(\Upsilon, \Sigma | Y) d\Sigma.$$

This matrix-variate integral can be avoided by simply writing

$$p(\Upsilon | Y) = \frac{p(\Upsilon, \Sigma | Y)}{p(\Sigma | \Upsilon, Y)}. \quad (21)$$

The density $p(\Upsilon, \Sigma | Y)$ comes from Equation (20), while the denominator is obtained as

$$p(\Sigma | \Upsilon, Y) = \frac{p(\Sigma | Y)p(\Upsilon | \Sigma, Y)}{p(\Upsilon | Y)} \\ \propto \frac{|\Psi^*|^{\frac{\nu^*}{2}} |V_{\gamma}^*|^{-\frac{q}{2}} |\Sigma|^{-\frac{\nu^*+m+q+1}{2}}}{2^{\frac{(\nu^*+m)q}{2}} (\pi)^{\frac{mq}{2}} \Gamma_q(\frac{\nu^*}{2})} \exp \left\{ -\frac{1}{2} \text{tr} [\Sigma^{-1} (\Psi^* + (\Upsilon - \mu^*)^{\top} V_{\gamma}^{*-1} (\Upsilon - \mu^*))] \right\} \\ \propto |\Sigma|^{-\frac{\nu^*+m+q+1}{2}} \exp \left\{ -\frac{1}{2} \text{tr} [\Sigma^{-1} (\Psi^* + (\Upsilon - \mu^*)^{\top} V_{\gamma}^{*-1} (\Upsilon - \mu^*))] \right\},$$

where $\mu^* = M\mu_\gamma^*$. Hence, $\Sigma \mid \Upsilon, Y \sim \text{IW}(\hat{\Psi}, \hat{\nu})$ with $\hat{\Psi} = (\Psi^* + (\Upsilon - \mu^*)^\top V_\gamma^{*-1} (\Upsilon - \mu^*))$, and $\hat{\nu} = \nu^* + m$. The joint posterior predictive density then follows from Equation (21)

$$\begin{aligned} p(\Upsilon \mid Y) &= \frac{\text{MNIW}(\Upsilon, \Sigma \mid \mu^*, V_\gamma^*, \Psi^*, \nu^*)}{\text{IW}(\Sigma \mid \hat{\Psi}, \hat{\nu})} \\ &= K(\Upsilon) |\Psi^*|^{-\frac{\nu^*+m}{2}} \left| \mathbb{I}_m + V_\gamma^{*-1} (\Upsilon - \mu^*) \Psi^{*-1} (\Upsilon - \mu^*)^\top \right|^{-\frac{\nu^*+m}{2}}, \end{aligned}$$

where $K(\Upsilon) = \frac{\Gamma_q(\frac{\hat{\nu}}{2}) |\Psi^*|^{\frac{\nu^*}{2}} |V_\gamma^*|^{-\frac{q}{2}}}{\Gamma_q(\frac{\nu^*}{2}) (\pi)^{\frac{mq}{2}}} = \frac{\Gamma_q(\frac{\nu^*+m}{2}) |\Psi^*|^{\frac{\nu^*}{2}} |V_\gamma^*|^{-\frac{q}{2}}}{\Gamma_q(\frac{\nu^*}{2}) (\pi)^{\frac{mq}{2}}}$, since $\hat{\nu} = \nu^* + m$.

This is a matrix-variate T density, which we denote as $\Upsilon \mid Y \sim \text{T}_{m,q}(\nu^*, \mu^*, V^*, \Psi^*)$. We recover the same result, without needing to integrate out Σ (Iranmanesh et al., 2010; Gupta and Nagar, 2000), by using only Bayes theorem and related distribution theory. Finally, the marginal predictive distributions $\Omega_{\mathcal{U}} \mid \mathcal{D}$ and $Y_{\mathcal{U}} \mid \mathcal{D}$ are also available in analytic form as matrix T distributions for any set of predictive points $\mathcal{U}$.

A.2 Details on Disagreement Tempering

Double Bayesian predictive stacking produces posterior predictive distributions that are expressed as mixtures. We can then take advantage of linear pooling (see, e.g., Knüppel and Krüger, 2022) to mitigate inflated predictive variances due to disagreements between the mixture and the true posterior. Unlike standard linear pooling, DBPS posterior predictive distributions are mixtures of a finite mixture distribution that includes a second layer of mixing. We adapt to this setting and show how disagreement tempering (DT) can mitigate over-dispersion in our model.

The DBPS posterior predictive distribution $\hat{p}(Y_{\mathcal{U}}; \mathcal{D})$ in (11) can also be written as

$$\hat{p}(Y_{\mathcal{U}}; \mathcal{D}) = \sum_{k=1}^K w_k \hat{p}(Y_{\mathcal{U}}; \mathcal{D}_k) = \sum_{k=1}^K w_k \sum_{j=1}^J z_{k,j} p(Y_{\mathcal{U}} \mid \mathcal{D}_k, \mathcal{M}_j).$$

Each component in the mixture is a matrix-variate T distribution $p(Y_{\mathcal{U}} \mid \mathcal{D}_k, \mathcal{M}_j) = \text{T}_{n',q}(\nu_k^*, \mu_{k,j}^*, V_{k,j}^*, \Psi_{k,j}^*)$ with degrees of freedom $\nu_k^* = \nu_0 + n_k$, an $n' \times q$ location matrix $\mu_{k,j}^* = M_{y,k,j} \mu_{\gamma,k,j}^*$, an $n' \times n'$ row-scale matrix $V_{k,j}^*$, and a column-scale matrix $\Psi_{k,j}^*$ of dimension $q \times q$, where all the parameters are defined in (6), (7), and (9).

Consider now the vectorization of $Y_{\mathcal{U}}$, i.e., $\tilde{Y}_{\mathcal{U}} = \text{vec}(Y_{\mathcal{U}})$, and denote $m_{k,j} = \text{vec}(\mu_{k,j}^*)$, $\bar{m}_k = \sum_{j=1}^J z_{k,j} m_{k,j}$ and $\bar{m} = \sum_{k=1}^K w_k \bar{m}_k$. Under the properties of the matrix-variate t distribution, $\tilde{Y}_{\mathcal{U}}$ is a multivariate t, and given that $\nu_k^* = \nu_0 + n_k > 2$ is satisfied under our prior choice $\nu_0 = 3$, its conditional covariance matrix is

$$\text{Var}(\tilde{Y}_{\mathcal{U}} \mid \mathcal{D}_k, \mathcal{M}_j) = \frac{\nu_k^*}{\nu_k^* - 2} \Psi_{k,j}^* \otimes V_{k,j}^*.$$

The posterior predictive covariance can then be derived as

$$\begin{aligned} \text{Var}(\tilde{Y}_{\mathcal{U}}; \mathcal{D}) &= \sum_{k=1}^K w_k \text{Var}(\tilde{Y}_{\mathcal{U}}; \mathcal{D}_k) + \sum_{k=1}^K w_k \bar{m}_k \bar{m}_k^\top - \bar{m} \bar{m}^\top \\ &= \sum_{k=1}^K w_k \left[\sum_{j=1}^J z_{k,j} \text{Var}(\tilde{Y}_{\mathcal{U}} \mid \mathcal{D}_k, \mathcal{M}_j) + \sum_{j=1}^J z_{k,j} m_{k,j} m_{k,j}^\top - \bar{m}_k \bar{m}_k^\top \right] + \sum_{k=1}^K w_k \bar{m}_k \bar{m}_k^\top - \bar{m} \bar{m}^\top. \end{aligned}$$

Expanding all terms and simplifying some of them, we get the complete formulation

$$\begin{aligned}
 &= \sum_{k=1}^K w_k \sum_{j=1}^J z_{k,j} \text{Var}(\tilde{Y}_{\mathcal{U}} \mid \mathcal{D}_k, \mathcal{M}_j) + \sum_{k=1}^K w_k \sum_{j=1}^J z_{k,j} m_{k,j} m_{k,j}^\top - \sum_{k=1}^K w_k \bar{m}_k \bar{m}_k^\top \\
 &\quad + \sum_{k=1}^K w_k \bar{m}_k \bar{m}_k^\top - \bar{m} \bar{m}^\top \\
 &= \sum_{k=1}^K w_k \sum_{j=1}^J z_{k,j} \text{Var}(\tilde{Y}_{\mathcal{U}} \mid \mathcal{D}_k, \mathcal{M}_j) + \sum_{k=1}^K w_k \sum_{j=1}^J z_{k,j} m_{k,j} m_{k,j}^\top - \bar{m} \bar{m}^\top \\
 &= \sum_{k=1}^K w_k \sum_{j=1}^J z_{k,j} \frac{\nu_k^*}{\nu_k^* - 2} \Psi_{k,j}^* \otimes V_{k,j}^* + \underbrace{\sum_{k=1}^K w_k \sum_{j=1}^J z_{k,j} m_{k,j} m_{k,j}^\top - \bar{m} \bar{m}^\top}_{\text{Dis}(\{m_{k,j}\}_{k,j})} \\
 &= \sum_{k=1}^K w_k \sum_{j=1}^J z_{k,j} \frac{\nu_k^*}{\nu_k^* - 2} \Psi_{k,j}^* \otimes V_{k,j}^* + \text{Dis}(\{m_{k,j}\}_{k,j}) .
 \end{aligned}$$

The disagreement term is the covariance matrix of the discrete random vector that takes value $m_{k,j} = \text{vec}(\mu_{k,j}^*)$ with probability $w_k z_{k,j}$: it measures how much the location matrices $\mu_{k,j}^*$ vary across the components of the mixture. As a covariance matrix, it satisfies $\text{Dis}(\{m_{k,j}\}_{k,j}) \succeq 0$, where $A \succeq 0$ denotes that a matrix A is positive semidefinite.

The idea of disagreement tempering, which can be considered a heuristic adjustment in machine learning, is to consider the individual posterior predictive distributions so that the new posterior predictive means are all equal to a constant, i.e., $\mathbb{E}_{\text{adjusted}}[Y_{\mathcal{U}} \mid \mathcal{D}_k, \mathcal{M}_j] = C$ for all k, j . If we are able to do this, then this will lead to $\text{Dis}(\{m_{k,j}\}_{k,j}) = 0$ and deflate the covariance matrix from the mixing. This “disagreement tempered” predictive covariance is

$$\sum_{k=1}^K w_k \sum_{j=1}^J z_{k,j} \frac{\nu_k^*}{\nu_k^* - 2} \Psi_{k,j}^* \otimes V_{k,j}^* = \text{Var}(\tilde{Y}_{\mathcal{U}} ; \mathcal{D}) - \text{Dis}(\{m_{k,j}\}_{k,j}) .$$

In practice, the centering is implemented as follows. For each $r = 1, \dots, R$, we sample $Y_{\mathcal{U}}^{(r)} \sim \text{T}_{n',q}(\nu_{k^{(r)}}^*, \mu_{k^{(r)},j^{(r)}}^*, V_{k^{(r)},j^{(r)}}^*, \Psi_{k^{(r)},j^{(r)}}^*)$ and record the corresponding posterior predictive mean $\mu_{k^{(r)},j^{(r)}}^* = \mathbb{E}[Y_{\mathcal{U}} \mid \mathcal{D}_{k^{(r)}}, \mathcal{M}_{j^{(r)}}]$ as the mean of the sampled matrix-variate t component. Each predictive draw is then centered by subtracting $\mu_{k^{(r)},j^{(r)}}^*$, and a common global mean $\mu_C^* = \frac{1}{R} \sum_{r=1}^R \mu_{k^{(r)},j^{(r)}}^*$ is added back to obtain the disagreement-tempered predictive samples. Algorithm 3 formalizes this procedure. Note that no explicit averaging over K and J is required, since the sampling at each iteration already accounts for the mixture weights, as is standard sampling from mixture models.

Appendix B. Computational Details

Key computational aspects of the proposed method involve two points: comparing its theoretical complexity to state-of-the-art approaches and addressing memory constraints. The complexity comparison evaluates the method’s time and space efficiency, particularly its scalability with larger datasets. This includes insights from the explicit objective function, highlighting its computational impact. Memory constraints are equally critical, as limita-

tions can hinder performance despite powerful processors. The proposed method addresses these challenges, ensuring both scalability and efficient resource use.

In summary, this section will examine both the theoretical complexity, including the explicit derivation of the objective function for the optimization problems detailed in Equations (8) and (22), as well as the memory management strategies, offering a comprehensive view of the computational feasibility in practical applications.

B.1 Objective Function for Double Bayesian Predictive Stacking

We expound the double Bayesian predictive stacking in Section 2.3. The optimization problem used to compute the stacking weights in Equation (10) is formally defined as:

$$\max_{w \in S_1^K} \frac{1}{n} \sum_{i=1}^n \log \sum_{k=1}^K w_k \hat{p}(Y_i; \mathcal{D}_k) = \max_{w \in S_1^K} \frac{1}{n} \sum_{i=1}^n \log \sum_{k=1}^K w_k \sum_{j=1}^J \hat{z}_{k,j} p(Y_{k,i} | \mathcal{D}_{k,[-l]}, \mathcal{M}_j), \quad (22)$$

as $\hat{p}(Y_i; \mathcal{D}_k) = \sum_{j=1}^J \hat{z}_{k,j} p(Y_{k,i} | \mathcal{D}_{k,[-l]}, \mathcal{M}_j)$. In this DOUBLE BPS framework, we focus exclusively on $\mathcal{D}_k$, which is treated equivalently to $\mathcal{M}_j$ in the first step. It is crucial to discriminate the predictive performance induced by each $\mathcal{D}_k$. We must use a common set of Y across all $\mathcal{D}_k$ —namely, Y itself. This stems from the construction of DOUBLE BPS. Specifically, for DOUBLE BPS to be effective, it necessitates predictive assessments over a common set of points for each model in the competition. Otherwise, the predictive performances cannot be directly compared, and the weights cannot be optimized to distinguish predictive capabilities across models, as different points would be used for different models.

To illustrate, consider the first stacking step. Here, we compute $p(Y_{k,i} | \mathcal{D}_k, \mathcal{M}_j)$ for each subset, where $i = 1, \dots, n_k$ and $j = 1, \dots, J$. This allows us to evaluate Y_k with respect to the predictive density of all J models under consideration. Similarly, in DOUBLE BPS, the goal is to evaluate Y_i with respect to the predictive density across all K subsets (acting as competing models) for comparison. The weights $\{\hat{z}_{k,j}\}$, which are derived from the optimization problem specified in Equation (8) also appear in Equation (22). However, comparing the right-hand sides of (22) and (8), we observe that the objective functions are almost identical, with the only difference being the second convex linear combination governed by the weights $\{w_k\}$. Therefore, the predictive distributions in both optimization problems refer to the same quantity. To summarize, the objective function in Equation (8) can be derived by substituting each $\{z_{k,j}\}$ with its optimized counterpart $\{\hat{z}_{k,j}\}$ and incorporating the weights $\{w_k\}$. This leads to the maximization objective in (22).

Next, we consider the optimization problem in (8) with the objective function,

$$\max_{z_k \in S_1^J} \frac{1}{n_k} \sum_{i=1}^{n_k} \log \sum_{j=1}^J z_{k,j} p(Y_{k,i} | \mathcal{D}_{k,[-l]}, \mathcal{M}_j) = \max_{z_k \in S_1^J} f(z_k),$$

where $f(z_k) = \frac{1}{n_k} \sum_{i=1}^{n_k} \log \sum_{j=1}^J z_{k,j} p(Y_{k,i} \mid \mathcal{D}_{k,[-l]}, \mathcal{M}_j)$. An explicit form of $f(z_k)$ is

$$\begin{aligned} f(z_k) &= f(z_{k,1}, \dots, z_{k,J}) = \frac{1}{n_k} \sum_{i=1}^{n_k} \log \sum_{j=1}^J z_{k,j} p(Y_{k,i} \mid \mathcal{D}_{k,[-l]}, \mathcal{M}_j) \\ &= \frac{1}{n_k} \sum_{i=1}^{n_k} \log \sum_{j=1}^J z_{k,j} T_{1,q}(Y_{k,i} \mid \nu_{[-l]}^*, \mu_i^*, V_i^*, \Psi_{[-l]}^*) \\ &= \frac{1}{n_k} \sum_{i=1}^{n_k} \log \sum_{j=1}^J z_{k,j} K(Y_{k,i}) \left| 1 + V_i^{*-1}(Y_{k,i} - \mu_i^*) \Psi_{[-l]}^{*-1}(Y_{k,i} - \mu_i^*)^T \right|^{-\frac{\nu_{[-l]}^*+1}{2}}, \end{aligned}$$

where $K(Y_{k,i}) = \frac{|\Psi_{[-l]}^*|^{-\frac{1}{2}} |V_i^*|^{-\frac{q}{2}} \Gamma_q\left(\frac{\nu_{[-l]}^*+1}{2}\right)}{(\pi)^{\frac{1q}{2}} \Gamma_q\left(\frac{\nu_{[-l]}^*}{2}\right)}$. The logarithm of a linear combination precludes

further accessibility, but $f(z_k)$ is computed easily by evaluating the matrix-T density. This is standard convex optimization (Yao et al., 2018); see Section 4 for further details.

The objective function in Equation (22) is related to Equation (8) as

$$\max_{w \in S_1^K} \frac{1}{n} \sum_{i=1}^n \log \sum_{k=1}^K w_k \sum_{j=1}^J \hat{z}_{k,j} p(Y_{k,i} \mid \mathcal{D}_{k,[-l]}, \mathcal{M}_j) = \max_{w \in S_1^K} g(w),$$

where $w = (w_1, \dots, w_K)^T$ and

$$\begin{aligned} g(w) &= \frac{1}{n} \sum_{i=1}^n \log \sum_{k=1}^K w_k \sum_{j=1}^J \hat{z}_{k,j} p(Y_{k,i} \mid \mathcal{D}_{k,[-l]}, \mathcal{M}_j) \\ &= \frac{1}{n} \sum_{i=1}^n \log \sum_{k=1}^K w_k \sum_{j=1}^J \hat{z}_{k,j} T_{1,q}(Y_{k,i} \mid \nu_{[-l]}^*, \mu_i^*, V_i^*, \Psi_{[-l]}^*) \\ &= \frac{1}{n} \sum_{i=1}^n \log \sum_{k=1}^K w_k \sum_{j=1}^J \hat{z}_{k,j} K(Y_{k,i}) \left| 1 + V_i^{*-1}(Y_{k,i} - \mu_i^*) \Psi_{[-l]}^{*-1}(Y_{k,i} - \mu_i^*)^T \right|^{-\frac{\nu_{[-l]}^*+1}{2}}. \end{aligned}$$

In practical applications, we address (22) separately from (8), which is defined for each subset. To evaluate the latter, we perform K separate maximizations in (8), one for each subset. Once we obtain all K sets of $\{\hat{z}_{k,j}\}$, we can recover the weights $\{\hat{w}_k\}$ across subsets by solving the convex optimization problem in (22).

Setting aside the potential for parallel computation of the subset stacking weights $\{\hat{z}_{k,j}\}$, this method offers a significant computational advantage: no additional quantities are required to solve the problem in Equation (22). All necessary components are already available from the independent computations performed within each subset. Specifically, all terms in Equations (8) and (22) are known and identical except for the weights w , which remain to be optimized. Consequently, there is no need to recompute the cross-validated predictive distributions or $\{\hat{z}_{k,j}\}$.

B.2 Theoretical Complexity

In terms of theoretical computational complexity, we provide a comparison between spatial meta-kriging (SMK, Guhaniyogi and Banerjee, 2018) and the DOUBLE BPS. Given a dataset $\mathcal{D}_n$, let n denote the total number of observations, K the number of subsets, and M the number of target posterior samples. The approach developed by Guhaniyogi and Banerjee (2018) generates each posterior sample through a parallel implementation of SMK over K cores, which requires $\mathcal{O}\left(\left(\frac{n}{K}\right)^3\right)$ operations. Thus, obtaining M draws from each subset posterior yields a complexity of $\mathcal{O}\left(M\left(\frac{n}{K}\right)^3\right)$ per computational core. The cost of computing the geometric median must also be added. As stated in Minsker et al. (2017), Weiszfeld’s algorithm has a complexity of $\mathcal{O}(M^2)$ per step and requires at most $\mathcal{O}(1/\epsilon)$ steps to achieve an accuracy within ϵ . However, especially in large-scale applications, allocating a computational core to each subset is not always possible. If we let m denote the number of available cores, where generally $m \ll K$, the total computational complexity of SMK across K partitions distributed over m cores becomes $\mathcal{O}\left(\frac{K}{m}\left[M\left(\frac{n}{K}\right)^3\right] + \frac{(KM)^2}{\epsilon}\right)$.

For the theoretical complexity of double Bayesian predictive stacking, we let J denote the number of competing models and L the number of folds used for cross-validation. Equivalent to SMK, model fitting within subsets is dominated by Cholesky decompositions, implying a cost on the order of $\mathcal{O}\left(\left(\frac{n}{K}\right)^3\right)$. However, in DOUBLE BPS, we perform J Cholesky decompositions and refit the model L times for each. Distributed over m cores, the theoretical complexity for this step is $\mathcal{O}\left(\frac{K}{m}JL\left(\frac{n}{K}\right)^3\right)$.

In addition, we find the stacking weights in polynomial time using an interior-point algorithm via the CVXR package (Fu et al., 2020) in the R statistical computing environment, which applies disciplined convex programming (Grant and Boyd, 2008; CVX Research, 2012). We use the solvers SCS (O’Donoghue et al., 2016) and ECOSolveR (Fu et al., 2023) to obtain these weights. Incorporating the disciplined convex programming steps into the total computational complexity yields $\mathcal{O}\left(\frac{K}{m}\left[JL\left(\frac{n}{K}\right)^3 + J^p\right] + K^p\right)$ for K subsets distributed over m cores, where p represents the polynomial degree of the solver.

The term in square brackets corresponds to the local operations within each subset, combining the cross-validation model fitting and the polynomial optimization cost across J models. The final K^p term accounts for the complexity introduced by the second stacking process across the K subsets.

Next, we compare the computational complexities of the two approaches. We will separately examine the terms associated with subset modeling and global inference combination. Thus, for SMK and DOUBLE BPS, respectively, the computational complexities are as follows:

$$\mathcal{O}\left(\underbrace{\frac{K}{m}\left[M\left(\frac{n}{K}\right)^3\right]}_{\text{subset modeling}} + \underbrace{\frac{(KM)^2}{\epsilon}}_{\text{combination}}\right), \quad \mathcal{O}\left(\underbrace{\frac{K}{m}\left[JL\left(\frac{n}{K}\right)^3 + J^p\right]}_{\text{subset modeling}} + \underbrace{K^p}_{\text{combination}}\right) \quad (23)$$

Focusing on the subset modeling component, as in (23), two specifications stand out. First, consider the difference in magnitude between M and the product JL . In this context, DOUBLE BPS offers a theoretical advantage when $JL < M$, a quite common condition in practice. This is because M represents the number of posterior samples required for convergence across all the Markov chains involved, and it typically needs to be at least on

the order of 10^3 . In contrast, the product JL consists of relatively small terms, making it highly likely that this inequality will hold. Second, due to the significant difference in scale, the term J^p is absorbed by $(n/K)^3$.

When comparing the combination phase, the analysis reduces to a comparison between the geometric median approximation and Bayesian predictive stacking. Since a discrete number of posterior samples M is required by SMK for each of the K partitions, we generally find that $K^p < (KM)^2/\epsilon$. Thus, while empirical computational times are significantly lower for double Bayesian predictive stacking compared to SMK, there are some modest theoretical differences between the two methods. The major advantage lies in avoiding simulation-based methods, such as MCMC while achieving local inferences through exact approaches.

Like Weiszfeld’s algorithm, modern disciplined convex programming encounters computational challenges in high-dimensional contexts, particularly in managing random memory allocation. In Section B.3, we present a feasible strategy for approximating the DOUBLE BPS weights, tailored for very large-scale memory problems.

B.3 Memory Management and Pseudo-BMA

When modeling GeoAI systems, as the number of locations exceeds the order of millions, managing storage space becomes crucial. Timing issues may arise depending on the available optimizer. While open-source solvers theoretically offer faster solutions compared to iterative algorithms, e.g., geometric median, they often face practical challenges when the problem size considerably exceeds dimensions of 10^2 . In contrast, commercial optimizers behave slightly better, even if these approaches are not exempt from random allocation memory constraints. We emphasize working with portable approaches, i.e., with open-source solvers, that can effectively handle large-scale problems.

We present a computationally cheaper alternative that facilitates better management of available RAM. The subsequent contents, including Algorithm 4, were implemented in data analyses involving millions (10^6) of locations in Section 6. When addressing optimization problems of significant dimensions, AIC-based alternatives could be considered.

To facilitate model stacking, various methodologies exist within the Bayesian model averaging (BMA) framework. In particular, we present an approach based on information criteria, which was formerly introduced in Yao et al. (2018). To ensure comparability between datasets and enhance interpretability, we estimate the expected log point-wise predictive density (as done in DOUBLE BPS). The expected log pointwise predictive density (*elpd*) for each partition k is defined as

$$\widehat{\text{elpd}}^k = \sum_{i=1}^n \widehat{\text{elpd}}_i^k = \sum_{i=1}^n \log \hat{p}(Y_{k,i} ; \mathcal{D}_{k,[-l]})$$

Importantly, each *elpd* term need not be computed individually, as these values are generated during the first BPS procedure within each subset for all model configurations, significantly reducing memory storage requirements and the total computational burden. Given the set $\{\widehat{\text{elpd}}^k\}_{k=1,\dots,K}$, the pseudo Bayesian model averaging (pseudo-BMA) weights are

$$\text{computed as } \hat{w}_k = \frac{\exp(\widehat{\text{elpd}}^k)}{\sum_{k=1}^K \exp(\widehat{\text{elpd}}^k)}.$$

Algorithm 4 Calculating stacking weights between subsets using pseudo-BMA

Input: $\hat{z} = \{\hat{z}_k = \{\hat{z}_{k,j}\} : k \in \{1, \dots, K\}, j \in \{1, \dots, J\}\}$: Stacking weights within subsets;
 $\{pd_{k,j,i} = T_{1,q}(Y_{k,i} \mid \nu_{[-l]}^*, \mu_i^*, V_i^*, \Psi_{[-l]}^*) : k = 1, \dots, K, j = 1, \dots, J, i = 1, \dots, n\}$: point-wise predictive density of Y ; n, q, p : Number of rows, number of outcomes, and number of predictors; $K, \{n_k : k \in \{1, \dots, K\}\}, J$: Number of subsets, dimension of each subset, and number of competitive models in each subset.

Output: $\hat{w} = \{\hat{w}_k : k = 1, \dots, K\}$: Stacking weights between subsets.

- 1: Construct $pd = \underbrace{[pd_1^T : \dots : pd_K^T]^T}_{n \times J}$, $pd_k = \underbrace{\begin{bmatrix} pd_{k,1,1} & \dots & pd_{k,J,1} \\ \vdots & pd_{k,j,i} & \vdots \\ pd_{k,1,n_k} & \dots & pd_{k,J,n_k} \end{bmatrix}}_{n_k \times J}$
- 2: **for** $k = 1, \dots, K$ **do**
- 3: Compute $\widehat{elpd}^k = \sum_{i=1}^n \log(pd \hat{z}_k)$
- 4: **end for**
- 5: **for** $k = 1, \dots, K$ **do**
- 6: Compute $\hat{w}_k = \exp(\widehat{elpd}^k) / \sum_{k=1}^K \exp(\widehat{elpd}^k)$
- 7: **end for**
- 8: **return** $\hat{w} = \{\hat{w}_k : k \in \{1, \dots, K\}\}$

This formulation, introduced by Yao et al. (2018), simplifies the computation of the stacking weights, significantly reducing computational costs in terms of complexity and storage while maintaining the Bayesian predictive stacking framework. Thus, it serves as a viable alternative to BPS for predictive densities in challenging scenarios. When dealing with datasets comprising millions of instances and a substantial number of partitions, optimization solvers may fail or produce errors due to memory constraints, and the iterative processes involving large matrices can lead to increased procedure times.

Based on empirical experience, we primarily use pseudo-BMA as a Bayesian predictive stacking approach when datasets require excessive memory storage, particularly when $n \gg 10^5$ and $K \gg 10^2$. When feasible, we generally prefer convex optimization using BPS of predictive densities without reservations. Simulations highlighting potential differences in posterior predictive and posterior inference performances between these two model stacking approaches can be found in Yao et al. (2018), where several alternatives to Bayesian stacking approaches are discussed. We opted for pseudo-BMA due to its simpler analytical formulation, which enables matrix algebra to mitigate the computational burden of both random allocation memory and runtime.

B.4 Memory-Efficient Posterior Sampling

In the matrix-variate conjugate Bayesian linear regression model presented in Equation (2), the Bayesian updating process may become costly when several data shards arrive, even more so when the dimensions involved are large. Computational problems are often related to the available RAM, especially when working with sizable datasets. We present a memory-efficient posterior sampling scheme for the regression coefficient β to reduce the

computational burden in such contexts. We consider the model in Equation (2) of Section 2.1,

$$\begin{aligned} Y &= X\beta + E, \quad E \mid \Sigma \sim \text{MN}(O, V, \Sigma); \\ \beta &= M_0 m_0 + E_\beta, \quad E_\beta \mid \Sigma \sim \text{MN}(O, M_0, \Sigma); \quad \Sigma \sim \text{IW}(\Psi_0, \nu_0), \end{aligned} \quad (24)$$

where Σ is assumed to be known hereafter. Then, by matrix normal distribution theory, we know the exact form of the posterior distribution

$$\beta \mid \mathcal{D}, \Sigma \sim \text{MN}(M_n m_n, M_n, \Sigma), \quad (25)$$

where $M_n^{-1} = M_0^{-1} + X^\top V^{-1} X$, $m_n = m_0 + X^\top V^{-1} Y$. We provide a memory-efficient way to sample from this distribution and reduce its computational burden. We define random variables $Y_{rep} \sim \text{MN}(Y, V, \Sigma)$ and $Z \sim \text{MN}(M_0 m_0, M_0, \Sigma)$. Expressing the relation between Y_{rep} , Z , and B as

$$M_n^{-1} B = A_1 Z + A_2 Y_{rep}. \quad (26)$$

We seek matrices A_1, A_2 such that $B \stackrel{d}{=} \beta \mid \mathcal{D}, \Sigma$. Since $\beta \mid \mathcal{D}, \Sigma$ is distributed as a Gaussian random variable, it is fully characterized by its mean and variance (in such cases, with both row and column covariance matrices). Then, all we need are A_1 and A_2 so that the first two moments of B match those of $\beta \mid \mathcal{D}, \Sigma$.

For $X \sim \text{MN}(m, v, s)$, we have $DXC \sim \text{MN}(DmC, DvD^\top, CsC^\top)$. Moreover, if X is $n \times q$, the row-variance matrix is defined as $v = \mathbb{V}_{row}(X) = \mathbb{E}[(X - m)(X - m)^\top] \text{tr}(s)^{-1}$ of dimension $(n \times n)$ and its elements are defined as the variance computed on each row, while the $q \times q$ column covariance matrix is denoted by $s = \mathbb{V}_{col}(X) = \mathbb{E}[(X - m)^\top(X - m)] \text{tr}(v)^{-1}$ (see, e.g., Gupta and Nagar, 2000, for further details). Hence, we compute the row covariance matrix for B (the column covariance matrix Σ is given). Note that

$$M_n^{-1} \mathbb{V}_{row}(B) M_n^{-1} = A_1 \mathbb{V}_{row}(Z) A_1^\top + A_2 \mathbb{V}_{row}(Y_{rep}) A_2^\top = A_1 M_0 A_1^\top + A_2 V A_2^\top \quad (27)$$

Setting these matrices as $A_1 = M_0^{-1}$, and $A_2 = X^\top V^{-1}$, we have

$$M_n^{-1} \mathbb{V}_{row}(B) M_n^{-1} = M_0^{-1} M_0 M_0^{-1} + X^\top V^{-1} V V^{-1} X = M_0^{-1} + X^\top V^{-1} X = M_n^{-1}. \quad (28)$$

This implies $\mathbb{V}_{row}(B) = M_n$. The mean follows from

$$M_n^{-1} \mathbb{E}[B] = A_1 \mathbb{E}[Z] + A_2 \mathbb{E}[Y_{rep}] = M_0^{-1} M_0 m_0 + X^\top V^{-1} Y = m_n, \quad (29)$$

and we obtain $\mathbb{E}[B] = M_n m_n$. Therefore, we can derive the next equality in distribution between B and the posterior distribution of the regression coefficient β as

$$B \stackrel{d}{=} \beta \mid \mathcal{D}, \Sigma \sim \text{MN}(M_n m_n, M_n, \Sigma). \quad (30)$$

This implies we can sample from $\beta \mid \mathcal{D}, \Sigma$ by solving a linear system. Specifically, by simply drawing samples from Z and Y_{rep} , we obtain a sample from $\beta \mid \mathcal{D}, \Sigma$ by solving the system $(M_0^{-1} + X^\top V^{-1} X)B = (M_0^{-1} Z + X^\top V^{-1} Y_{rep})$ for B . This approach is particularly advantageous for Bayesian transfer learning, as it avoids storing several large matrices when computing the posterior of $\beta \mid \mathcal{D}, \Sigma$. Instead, only the prior precision matrix for β , M_0^{-1} , and the product matrix $X^\top V^{-1}$ need to be stored, significantly reducing the memory footprint.

Appendix C. Simulations for Multivariate Models

We supplement empirical results from Section 5; for further details, we refer to the main article. The current section follows this structure: we begin by comparing the computational performance of the double Bayesian predictive stacking (DBPS) approach with the multivariate spatial meta-kriging (MSMK, Guhaniyogi and Banerjee, 2019), the linear model of coregionalization (LMC, Finley et al., 2015), and the seemingly unrelated BART model (suBART, Esser et al., 2025), and conclude with empirical investigations for data partition dimension sensitivity for the DOUBLE BPS.

C.1 Computational Performance

We investigate the running times of our framework on two synthetic datasets with similar structures but different sizes. Both datasets consist of $p = 2$ predictors and $q = 2$ response variables, but differ in the number of spatial locations, $n = 5,000$ and $n = 10,000$, respectively. We generate and fix spatial coordinates from a uniform distribution on the unit square $([0, 1]^2)$. We build the $n \times n$ spatial correlation matrix V over these coordinates using $\rho_\phi(s_i, s_j) = \exp(-\phi \|s_i - s_j\|)$ with $\phi = 4$ and specify $\Sigma = \mathbb{I}_q$. From these specifications we generate the $n \times q$ matrix Y from the first equation of (4) with fixed $p \times q$ matrix $\beta = \begin{bmatrix} -0.75 & 1.85 \\ 0.90 & -1.10 \end{bmatrix}$, a fixed $n \times p$ matrix X with a first column of ones, representing the intercept, and $p - 1$ columns of values randomly simulated from a uniform distribution on $[0, 1]$ (emulating standardized predictors), and the proportion of spatial variability $\alpha = 0.8$.

For distributed learning approaches, i.e., DOUBLE BPS and MSMK, we perform the analyses in two settings: (i) $K = 10$ and $n = 5,000$; (ii) $K = 10$ and $n = 10,000$; (iii) $K = 5$ and $n = 5,000$; and (iv) $K = 20$ and $n = 10,000$. These settings produce subsets of size $n/K \in \{500, 1000\}$. We implement DOUBLE BPS using $J = 9$ candidate models $\mathcal{M}_j, j = 1, \dots, J$, where each model is specified by a set of candidate values for the hyperparameters α_j, ϕ_j in (4). These hyperparameters represent the proportion of spatial variability and the parameter(s) of the spatial correlation function, respectively. The set of candidate models is constructed as the set of all possible combinations of values for these hyperparameters. In the subsequent experiments, the grid of models was built using $\alpha \in \{0.70, 0.80, 0.90\}$ and $\phi \in \{3, 4, 5\}$. These values resemble an effective spatial range of $\{0.99, 0.75, 0.60\}$ units, corresponding to 70%, 53%, 42% of the maximum inter-site distance inside the unit square, beyond which the spatial correlation drops below 0.05. Equation (5) follows choices in Zhang et al. (2025). Specifically, we set $m_0 = 0_{p \times q}$, $M_0 = 10\mathbb{I}_p$, $\Psi_0 = \mathbb{I}_q$, and $\nu_0 = 3$ in the MNIW joint prior for $\{\gamma, \Sigma\}$ in 5. For data analysis, we use an exponential spatial correlation function for $\rho_\phi(\cdot, \cdot)$, which is completely defined conditionally on $\mathcal{M}_j$, since it specifies a value for ϕ . Again, in the conjugate framework, we draw $R = 250$ posterior samples used for inference.

We also apply MSMK to the two simulated datasets with the same combinations of n and K as DOUBLE BPS. Unlike DOUBLE BPS, where we stack analytically tractable posteriors over a range of fixed values of spatial covariance kernel parameters, the MSMK implementation attempts full Bayesian inference using prior distributions on spatial covariance kernel parameters. We fit the linear model of coregionalization described in Finley et al. (2015) for each subset of the multivariate spatial data using MCMC. The posterior samples from the K subsets are combined using Weiszfeld’s iterative algorithm (Minsker et al., 2017)

Setting	Model	Time (min)	Relative to BPS	Visual
$n = 5000, K = 10$	DBPS	0.24	1.0×	
	MSMK	51.41	220.3×	
$n = 5000, K = 5$	DBPS	0.98	1.0×	
	MSMK	237.23	241.2×	
$n = 5000$	DBPS	0.23 – 0.98	1.0 – 1.0×	–
	LMC	8975.31	>> 10000×	
	suBART	215.67	924 – 219×	 – 
$n = 10000, K = 20$	DBPS	0.42	1.0×	
	MSMK	103.36	248.1×	
$n = 10000, K = 10$	DBPS	1.92	1.0×	
	MSMK	446.01	232.7×	
$n = 10000$	DBPS	0.42 – 1.92	1.0 – 1.0×	–
	LMC	–	–	–
	suBART	557.60	1338 – 290×	 – 

Table 5: Running times (in minutes), relative to DOUBLE BPS. Bars give a visual impression of time cost (where applicable).

to produce an estimate of the geometric median of the posterior distributions. For both experimental settings, we then fit LMC and suBART on the full dataset, using default prior settings, following Finley et al. (2015), and Esser et al. (2025), respectively.

Table 5, which compares the computational speed of DOUBLE BPS with other considered approaches, reveals massive computational gains accrued from DOUBLE BPS. The computational advantage evident from the relative ratio becomes more pronounced as the size of the data increases, despite the larger subsets. This is explained by the fact that fitting the Gaussian process regression dominates the computation relative to assimilating inference from the subsets. If the number of locations explodes, then the geometric median of posteriors required by MSMK is computationally unfeasible. While MSMK offers Bayesian estimates using MCMC for each subset, DOUBLE BPS avoids MCMC and, hence, issues of convergence. Similar arguments follow for LMC and suBART. As expected, the linear model of coregionalization, when fitted on the entire set of locations, gives a disastrous performance, taking almost a week of computation in the lighter simulation settings, and makes it infeasible to record results for $n = 10,000$. Notwithstanding the scalability offered by this multivariate extension of the Bayesian additive regression model, the suBART does not have any chance to provide inference in a comparable time.

Figure 8, which depicts estimated response surfaces using DOUBLE BPS and MSMK corresponding to $n = 5000$ and $K = 10$, shows that inferences are practically indistinguishable. The root mean squared prediction error (RMSPE), reported in Figure 8, denotes the average prediction error and reveals minor discrepancies between DOUBLE BPS and MSMK. Section C presents results for the other configurations of n and K , each of which reveals that DOUBLE BPS offers practically indistinguishable spatial interpolation from MSMK at a fraction of the computational cost.

Figure 9 reports 95% posterior predictive intervals for the response. The empirical coverage is impressive. We notice slightly narrower intervals from DOUBLE BPS, indicating a better calibration of uncertainty quantification with respect to MSMK. Moreover, Figure 9

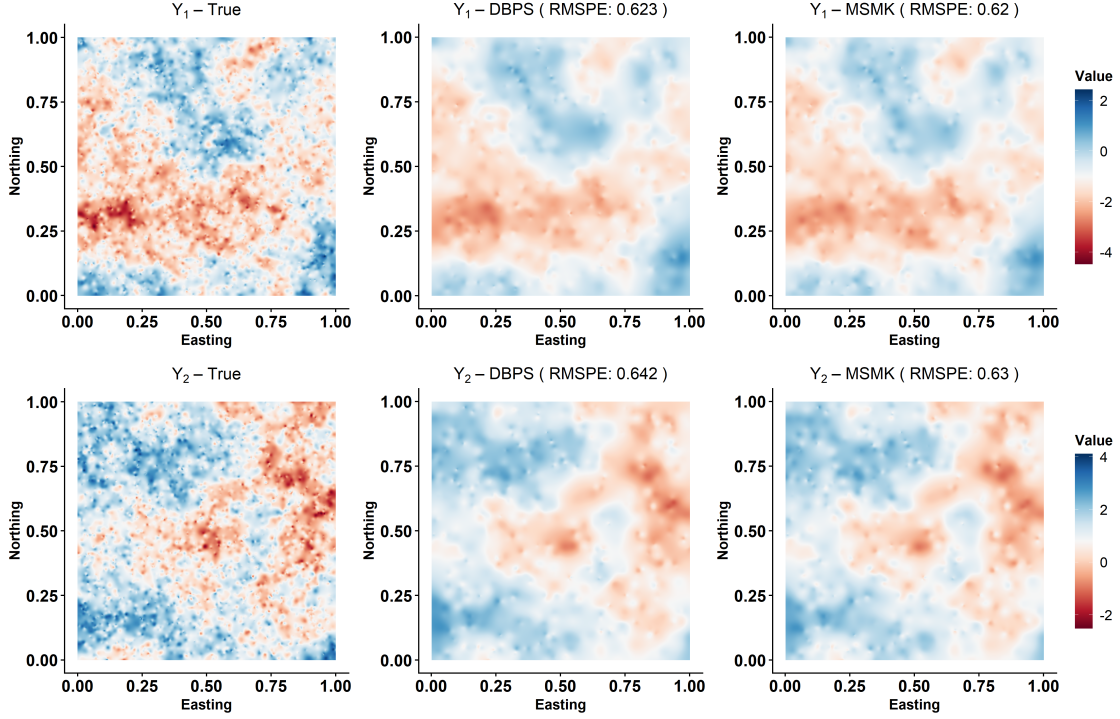


Figure 8: From left to right: comparison between the true generated response surfaces, the surfaces predicted from DOUBLE BPS and MSMK (posterior mean), with RMSPE. For $n = 5000$, $K = 10$.

reveals superior MAP estimates for the DOUBLE BPS. Finally, Figure 10 presents the recovery of parameter estimates. As seen in predictive inference, the posterior credible intervals for parameters also deliver practically indistinguishable inference for the two modeling frameworks. In particular, both methods recover the true values for β and Σ , while DOUBLE BPS reconstructs a better point estimate for range parameters ϕ using $\sum_{k=1}^K \hat{w}_k \sum_{j=1}^J \hat{z}_j \phi_j$.

Figure 11 shows the estimated response surface using DOUBLE BPS for $\{n = 5000, K = 5\}$. This plot reveals no noteworthy differences compared to the results obtained from MSMK. Although the RMSPE values for DBPS are slightly better, the differences are negligible when compared with the response scale, leading us to conclude that both approaches exhibit almost identical predictive performance. Regarding the first scenario from Section C.1, it is unsurprising that a doubled number of locations per partition leads to improved performance. Figure 13 provides empirical support to this. While MSMK shows some degree of spatial over-smoothing, as indicated by higher estimated values of ϕ compared to the true ones, DOUBLE BPS does not exhibit such deviations. Both the MAP estimates and the posterior inferences for DOUBLE BPS show strong performance.

Figure 12 aligns with the previous conclusions, highlighting slightly wider predictive credibility intervals for MSMK compared to DOUBLE BPS. For both approaches, the empirical coverage remains impressive.

Despite focusing on the larger dataset later in this section, where $n = 10,000$, the results increasingly appear to depend on the size of the subsets rather than the number of

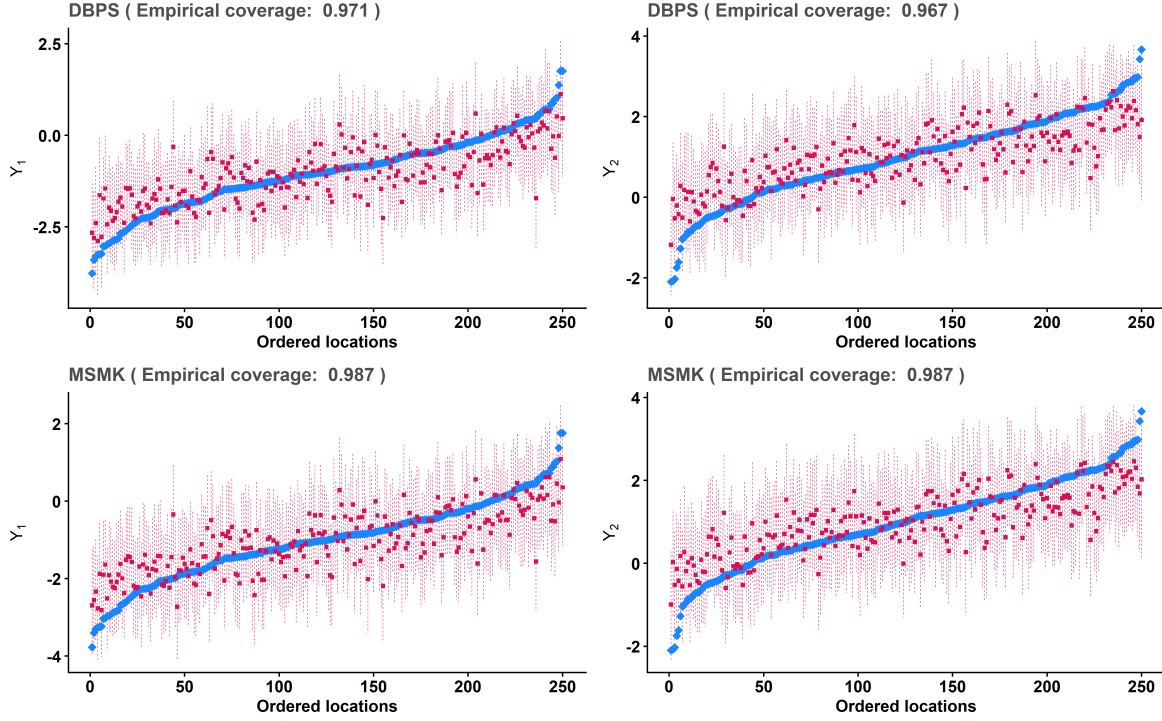


Figure 9: From top to bottom: comparison between posterior predictive intervals for the predicted response from DOUBLE BPS and MSMK, with empirical coverage. For $n = 5000$, $K = 10$.

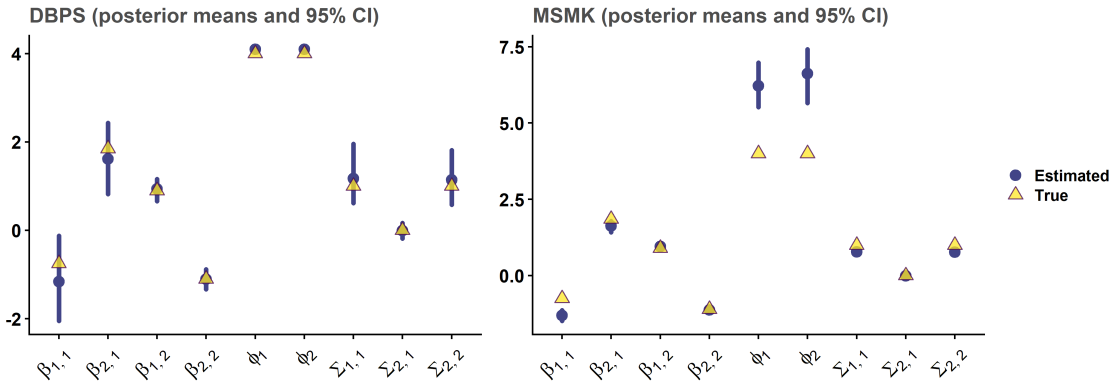


Figure 10: From left to right: comparison between posterior credible intervals for the parameters recovered from DOUBLE BPS and MSMK. For $n = 5000$, $K = 10$.

partitions. This highlights the influence of local inferences on global performance, with the number of partitions playing a lesser role.

For the case $\{n = 10,000, K = 20\}$, Figure 14 compares the estimated posterior mean surfaces. The performances of DOUBLE BPS and MSMK are nearly equivalent. The most noticeable difference is in Figure 15, where MSMK produces wider predictive credible intervals for the response variable. However, in Figure 16, DOUBLE BPS demonstrates su-

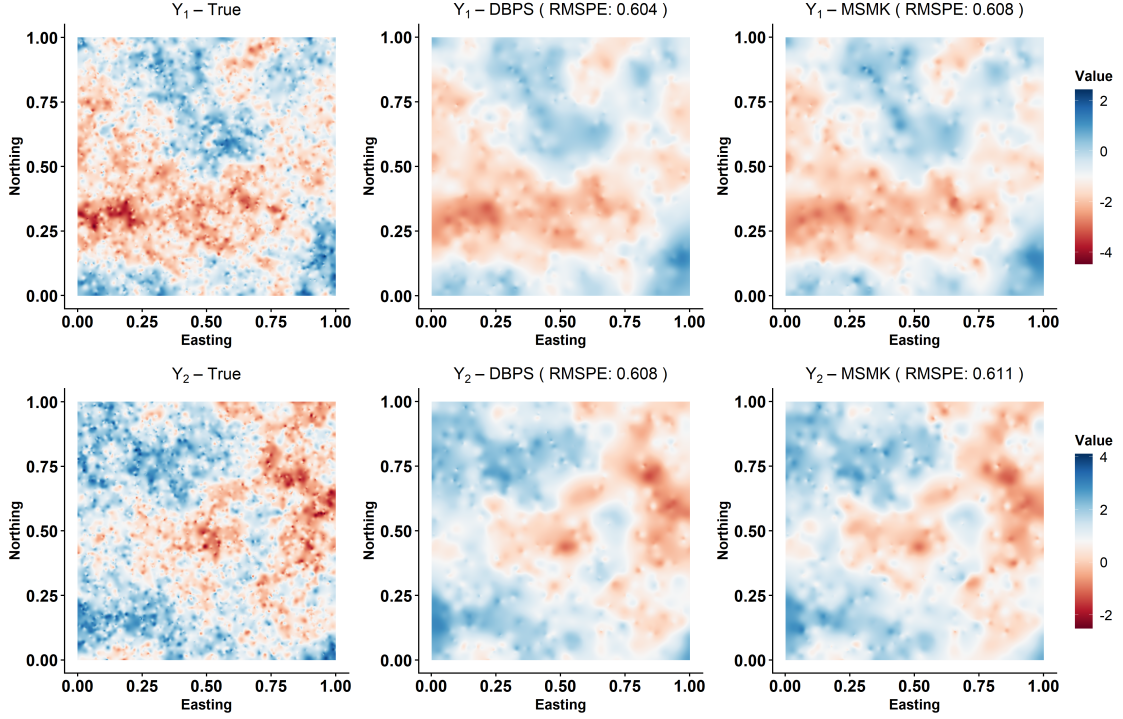


Figure 11: From left to right: comparison between the true generated response surfaces, the surfaces predicted from DOUBLE BPS and MSMK (posterior mean), with RMSPE. For $n = 5000$, $K = 5$.

prior posterior MAP estimates, including for the range parameters, which are typically challenging to identify. Together, Figures 15, 16, and 14 present comparable findings to the $\{n = 5,000, K = 10\}$ setting, reinforcing the importance of subset learning over the overall dataset size, as expected.

The simulation experiment concludes with the $\{n = 10,000, K = 10\}$ setting. Once again, Figures 17, 18, and 19 show results consistent with previous settings, particularly with $\{n = 5,000, K = 10\}$, highlighting the critical role of partition size in achieving reliable posterior inferences. In Figure 19, despite the stronger posterior performance, DOUBLE BPS exhibits wider posterior credible intervals compared to MSMK. Conversely, in Figure 18, DOUBLE BPS achieves predictive empirical coverage closer to the nominal level 95%.

In conclusion, notwithstanding indistinguishable posterior inferences between DBPS and multivariate SMK, Table 5 clearly illustrates that the primary advantage of Bayesian predictive stacking lies in its enormous computational efficiency. This speedup is crucial for delivering feasible Bayesian inference for large datasets within GeoAI systems.

C.2 Monte Carlo Approximation for Upper Bound Simulations

We perform empirical investigations of the upper bound presented and detailed in Section 3.1, for different values of K and J ceteris paribus.

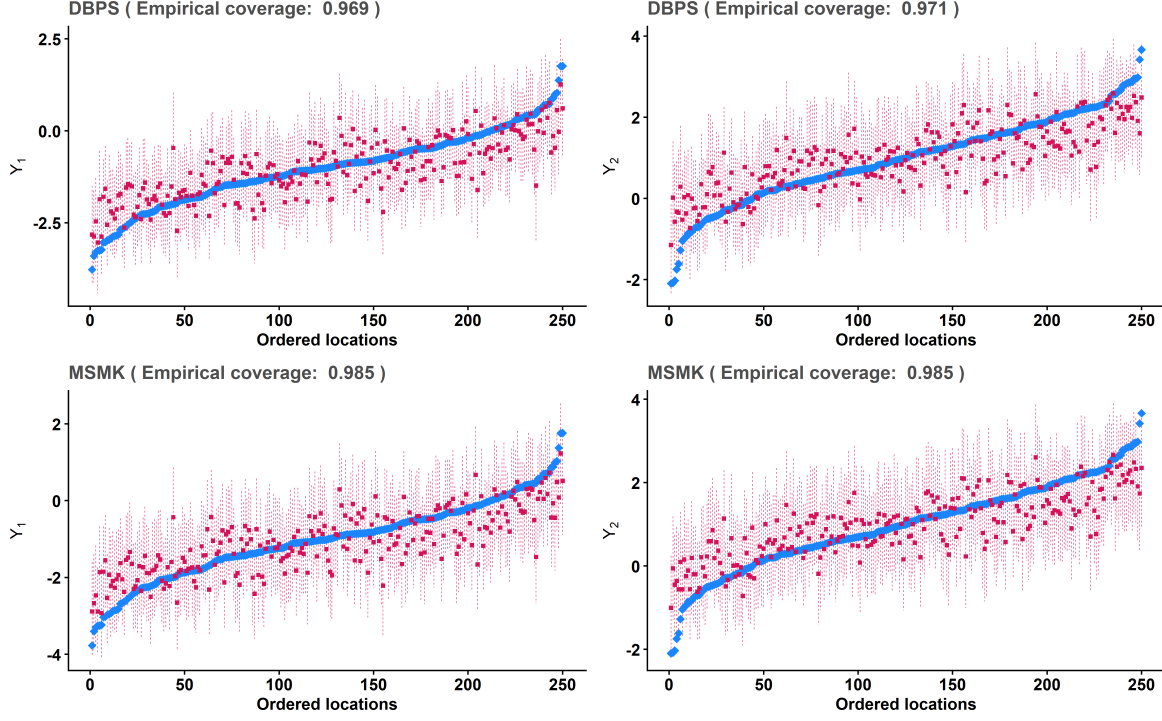


Figure 12: From top to bottom: comparison between posterior predictive intervals for the predicted response from DOUBLE BPS and MSMK, with empirical coverage. For $n = 5000$, $K = 5$.

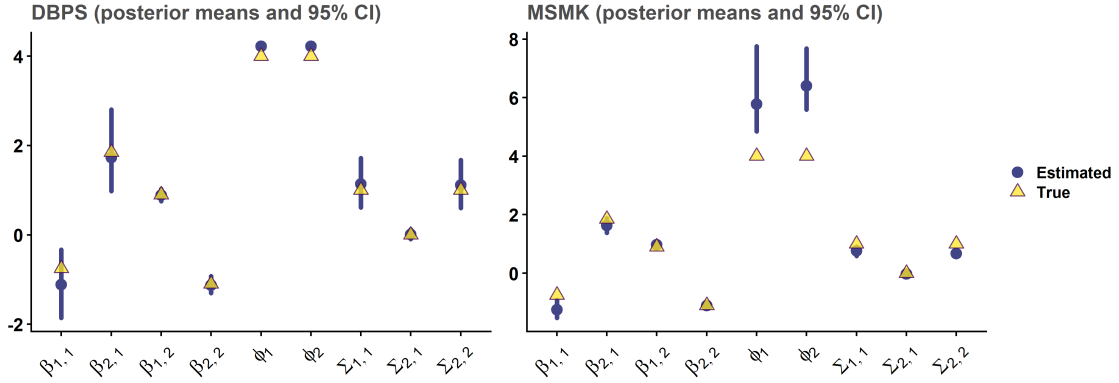


Figure 13: From left to right: comparison between posterior credible intervals for the parameters recovered from DOUBLE BPS and MSMK. For $n = 5000$, $K = 5$.

We approach the problem of approximating the expectation in Equation (12) with a Monte Carlo integration. The approximation takes the form

$$\mathbb{E}_{p_{k,j}} \left[\frac{\sum_{j=1}^J \hat{z}_{k,j} p(y | \mathcal{D}_k, \mathcal{M}_j)}{p_t(y | \mathcal{D})} \right] \approx \frac{1}{L} \sum_{l=1}^L \left[\frac{\sum_{j=1}^J \hat{z}_{k,j} p(y_l | \mathcal{D}_k, \mathcal{M}_j)}{p_t(y_l | \mathcal{D})} \right],$$

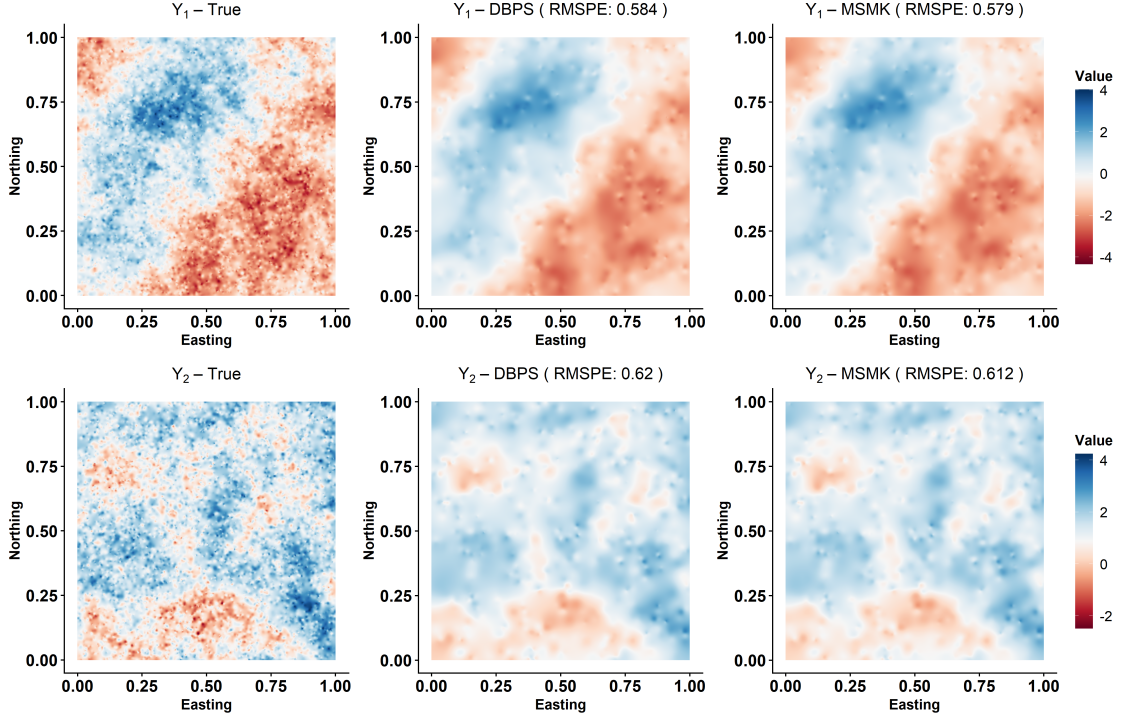


Figure 14: From left to right: comparison between the true generated response surfaces, the surfaces predicted from DOUBLE BPS and MSMK (posterior mean), with RMSPE. For $n = 10000$, $K = 20$.

where $y_l \sim p(y_l \mid \mathcal{D}_k, \mathcal{M}_j)$ for $l = 1, \dots, L$. We then devise the Algorithm 5 to approximate the upper bound for the KL divergence between the DOUBLE BPS posterior predictive and the true one.

To provide a meaningful interpolation, we consider 20 points for each parameter regulating $ub(n, K, J)$. We let vary $K \in \{5, 100\}$, $J \in \{2, 40\}$, while $n = 1000$ was fixed. Then, we remove data dependency by considering $M = 10$ replications for each evaluation setting. We perform different simulations for any of $\{K, J\}$ ceteris paribus, for the other. The panels in Figure 20 show how $D_{KL}(\hat{P} \parallel P_t)$ vary with K , and J , respectively.

C.3 Subset Size Sensitivity

The methodological novelty introduced in Section 2.3 can be summarized in three main steps, as illustrated in Figure 1. First, we partition the original, often massive, dataset into K smaller subsets. The number of locations in each partition is a critical decision, seriously impacting inferential, predictive, and computational outcomes. Accordingly, a trade-off arises between computational resources and performance.

To address this, we conduct a simulation analysis to assess the sensitivity of the results to subset size. This section aims to investigate how predictive performance (in terms of RMSPE), and runtime (in seconds) change as the number of locations within each partition grows. Intuitively and theoretically, as the dimension of the subsets grows, we expect predictive performance to improve, while runtime increases polynomially with n . To enhance

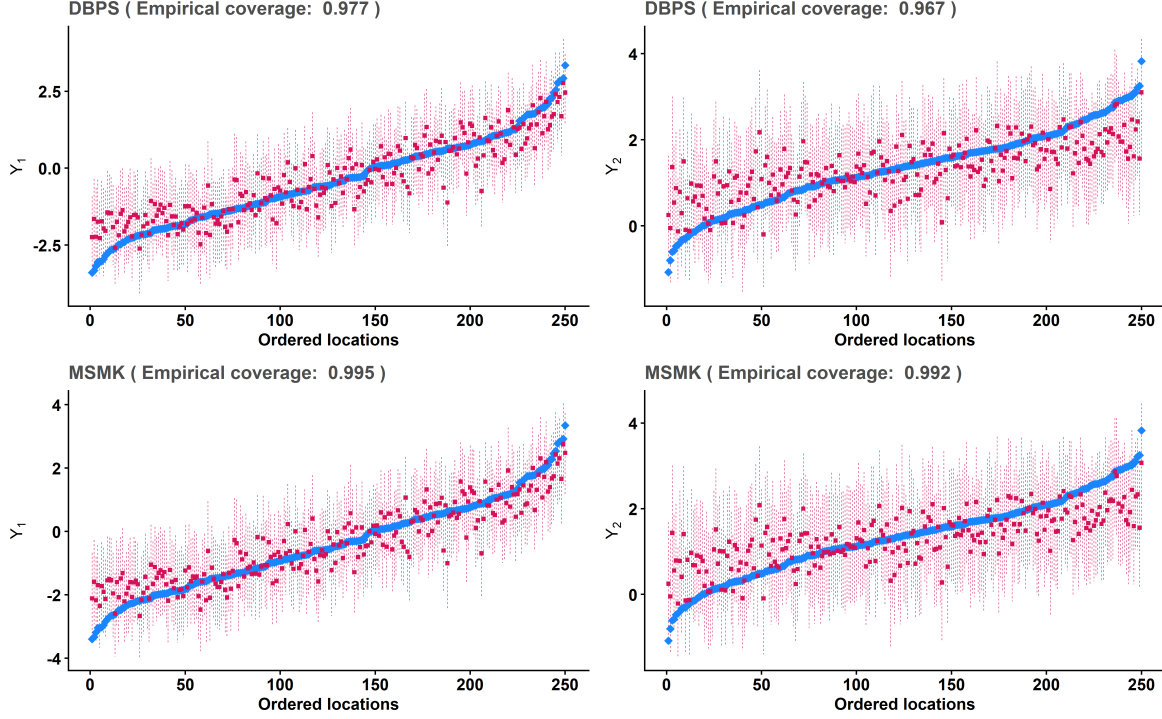


Figure 15: From top to bottom: comparison between posterior predictive intervals for the predicted response from DOUBLE BPS and MSMK, with empirical coverage. For $n = 10000$, $K = 20$.

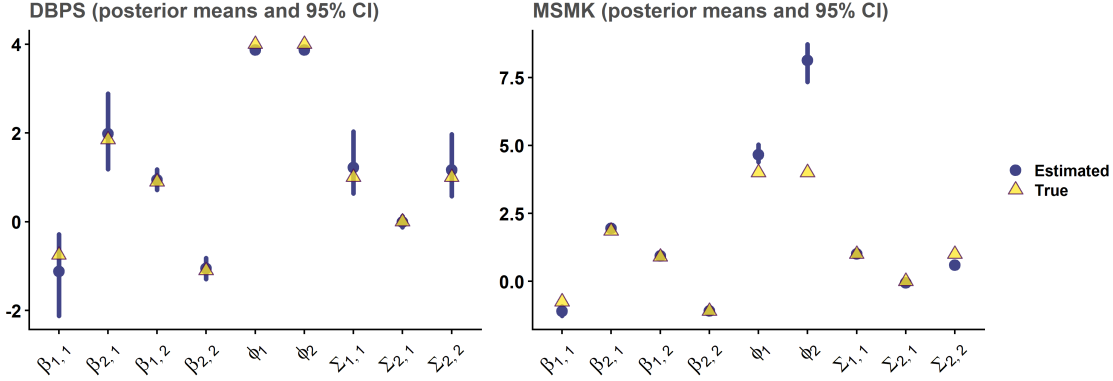


Figure 16: From left to right: comparison between posterior credible intervals for the parameters recovered from DOUBLE BPS and MSMK. For $n = 10000$, $K = 20$.

the comparability of the results, we apply min-max normalization to each variable, defined as $\tilde{x} = \frac{x - \min(x)}{\max(x) - \min(x)}$. This normalization scales all variables to the interval $[0, 1]$, facilitating a more direct graphical comparison.

We use a multivariate synthetic dataset comprising $n = 5,000$ locations, $q = 2$ simulated responses, and $p = 2$ predictors to explore the sensitivity to subset size. This dataset is generated from the model in Equation (4), with parameters $\beta = \begin{bmatrix} -0.75 & 1.85 \\ 0.90 & -1.10 \end{bmatrix}$ and $\Sigma =$

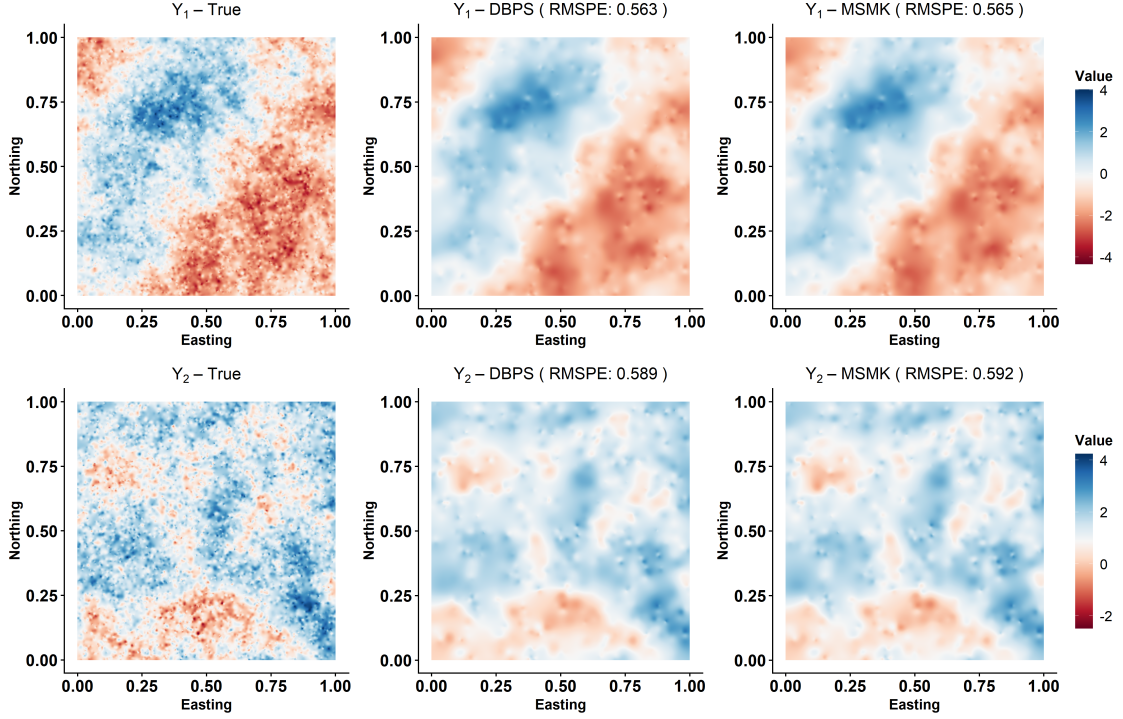


Figure 17: From left to right: comparison between the true generated response surfaces, the surfaces predicted from DOUBLE BPS and MSMK (posterior mean), with RMSPE. For $n = 10000$, $K = 10$.

$\begin{bmatrix} 1.00 & -0.30 \\ -0.30 & 1.00 \end{bmatrix}$. The predictor matrix X includes an intercept and $p - 1$ columns generated from a standard uniform distribution over $[0, 1]$. The range parameter for the exponential spatial covariance function, and the proportion of spatial variability, are fixed at $\phi = 4$ and $\alpha = 0.8$, respectively. We set prior information as follows: $m_0 = 0_{p \times q}$, $M_0 = 10\mathbb{I}_p$, $\Psi_0 = \mathbb{I}_q$, and $\nu_0 = 3$. These specifications and prior information remain constant, allowing only the number of locations in each subset to vary. In performing DOUBLE BPS detailed in Section 2.3, we consider $J = 9$ competitive models characterized by $\alpha \in \{0.70, 0.80, 0.90\}$ and $\phi \in \{3, 4, 5\}$.

We focus our sensitivity analysis by selecting the following set of partition sizes: $\{25, 50, 100, 250, 500, 1000, 1250\}$, which correspond to the number of partitions $K \in \{200, 100, 50, 20, 10, 5, 4\}$. Figure 21 illustrates the two curves resulting from this sensitivity analysis.

As anticipated, the behavior of the two curves aligns with theoretical expectations across most scenarios. Specifically, the total time required to fit the model increases monotonically, exhibiting more than linear growth in the number of locations within each partition, as shown by the dashed line in Figure 21. Conversely, the root mean square prediction error (RMSPE) decreases with partition size until it stabilizes at a “plateau” from approximately 500 units onward. The intersection of the two normalized curves in Figure 21 identifies a trade-off between predictive performance and computational effort close to a subset size of approximately 500 units.

However, Figure 21 also raises an important question: how much predictive error is acceptable? The two quantities, although normalized for comparison, differ significantly in

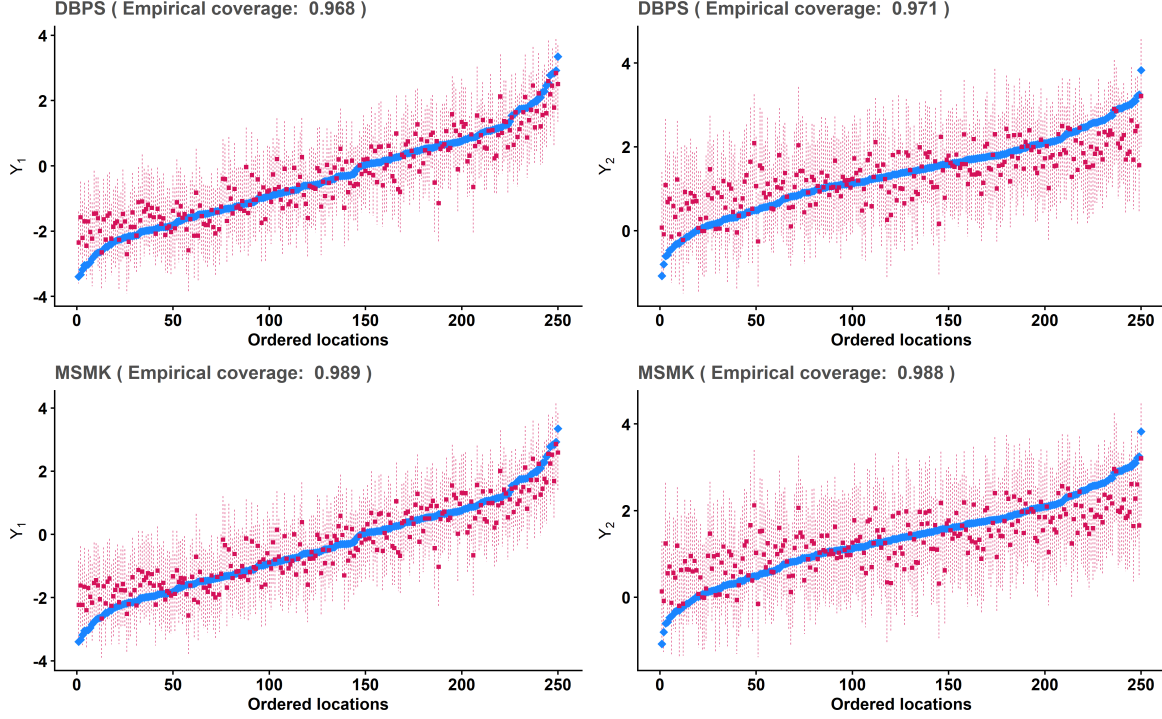


Figure 18: From top to bottom: comparison between posterior predictive intervals for the predicted response from DOUBLE BPS and MSMK, with empirical coverage. For $n = 10000$, $K = 10$.

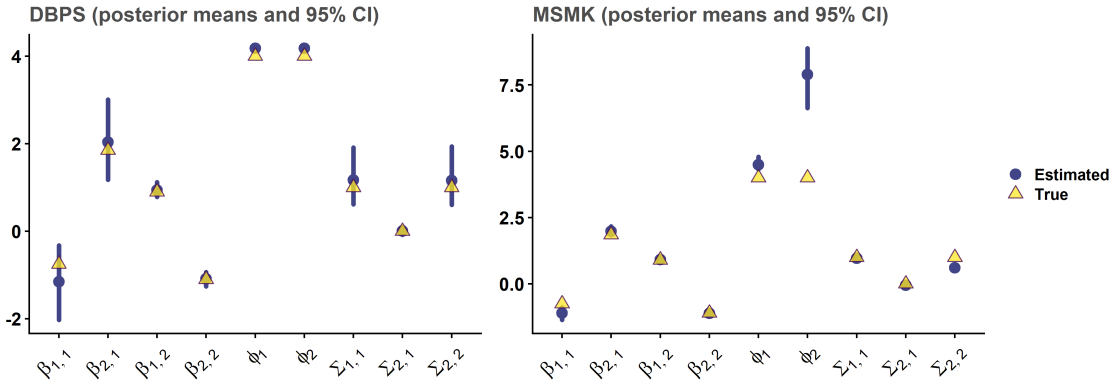


Figure 19: From left to right: comparison between posterior credible intervals for the parameters recovered from DOUBLE BPS and MSMK. For $n = 10000$, $K = 10$.

their scales. More precisely, the trade-off is asymmetric: doubling the number of locations per partition yields a moderate reduction in RMSPE, while the runtime can increase dramatically, rising at least quadratically with n . For all these reasons, we generally opt for a subset size of 500 locations in both our simulation studies and data applications. Nonetheless, we should not overlook the opportunity to reduce this size, accepting a compromise

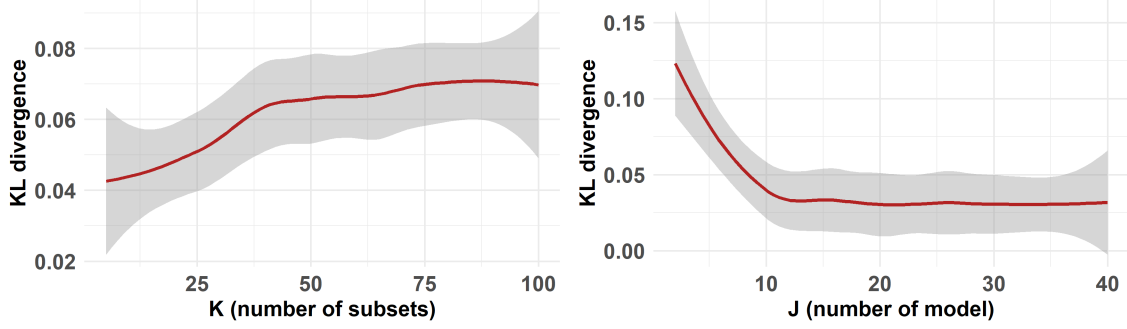


Figure 20: Upper bound behavior for growing values of K , and J

Algorithm 5 Approximating upper bound for Kullback-Liebler divergence

Input: Y outcomes matrix; X predictors matrix; $\hat{w} = \{\hat{w}_k : k = 1, \dots, K\}$: Stacking weights between subsets; $\hat{z} = \{\hat{z}_k = \{\hat{z}_{k,j} : k \in \{1, \dots, K\}, j \in \{1, \dots, J\}\}$: Stacking weights within subsets; $\hat{p}_{k,j}(\cdot), p_t(\cdot)$ approximated and true predictive distributions $\forall j = 1, \dots, J, k = 1, \dots, K$; K : Number of subsets; J : number of competitive models in each subset; n : number of locations; L : number of Monte Carlo samples.

Output: $\widehat{ub}(n, K, J)$: approximated value for the upper bound, for a given set $\{n, K, J\}$.

```

1: for  $k = 1, \dots, K$  do
2:   for  $j = 1, \dots, J$  do
3:     Draw  $L$  samples  $\{y_l : l = 1, \dots, L\}$  from  $\hat{p}(\cdot | \mathcal{D}_k, \mathcal{M}_j)$ 
4:     for  $l = 1, \dots, L$  do
5:       for  $j = 1, \dots, J$  do
6:         Evaluate  $p_{k,j,l} = \hat{p}(y_l | \mathcal{D}_k, \mathcal{M}_j)$ 
7:       end for
8:       Evaluate  $p_{t,l} = p(y_l | \mathcal{D})$ 
9:       Compute  $r_l = \frac{\sum_{j=1}^J \hat{z}_{k,j} p_{k,j,l}}{p_{t,l}}$ 
10:    end for
11:    Compute  $e_{k,j} = \frac{1}{L} \sum_{l=1}^L r_l$ 
12:  end for
13: end for
14: Compute  $c_k = \sum_{k=1}^K \hat{w}_k \sum_{j=1}^J \hat{z}_{k,j} e_{k,j}$ 
15: return  $\widehat{ub}(n, K, J) = \log \prod_{k=1}^K c_k^{\hat{w}_k}$ 

```

in predictive performance to achieve even faster global Bayesian inference for exceptionally large GeoAI applications.

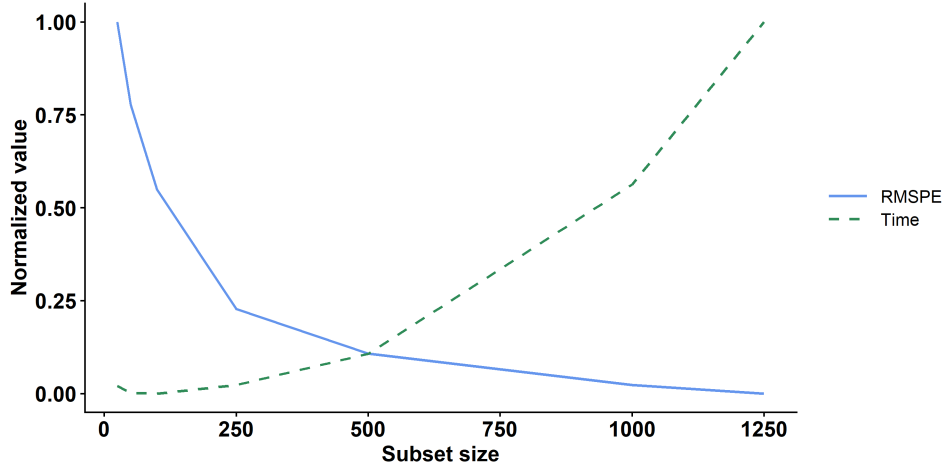


Figure 21: Comparison between average RMSPE (solid line) and model fitting time (dashed line) across various subset dimensions (both min-max normalized).

Σ_{NDVI}	Σ_{RR}	$\rho_{\text{NDVI,RR}}$
0.2208 (0.2202, 0.2215)	0.1549 (0.1545, 0.1553)	-0.9049 (-0.9052, -0.9046)

Table 6: Non-spatial association between response variables. 50 (2.5, 97.5) quantile estimates using Bayesian multivariate linear regression.

Appendix D. Exploratory Data Analysis

We illustrate more exploratory analysis on Vegetation Index data presented in Section 6.1. This section starts from the model-based non-spatial association among response variables, and concludes by presenting machine-generated supportive exploratory data insight used in Section 6.

We investigate non-spatial association between NDVI and red reflectance fitting the Bayesian multivariate regression model, defined by Equation (2). The model comprises two predictors: an intercept and the solar zenith angle for the million locations in the training set. More details on modeling and prior distribution are provided in Section 6, where a comparison of predictive performances is presented. Table 6 reports quantile estimates for marginal variances, i.e., the diagonal elements of Σ , and the correlation between NDVI and RR. Strong negative values for correlation are estimated, showing an intense inverse relationship between the two spatially dependent outcomes.

Hereafter, we present results from the fully automated exploratory spatial data analyses that complement Section 6. Variograms are employed in both analyses to extract “guidelines” on spatial parameters such as spatial variability proportion α , and spatial range ϕ , which are essential for setting up the DOUBLE BPS framework for GeoAI applications. Variogram fitting, used to gather parameter values required for DOUBLE BPS, is fully automated and requires no human intervention, except for specifying the grid width.

First, we use independent sample variograms for NDVI and RR, based on 31,875 randomly sampled locations. For NDVI, the empirical variogram estimates the nugget 0.03, a sill

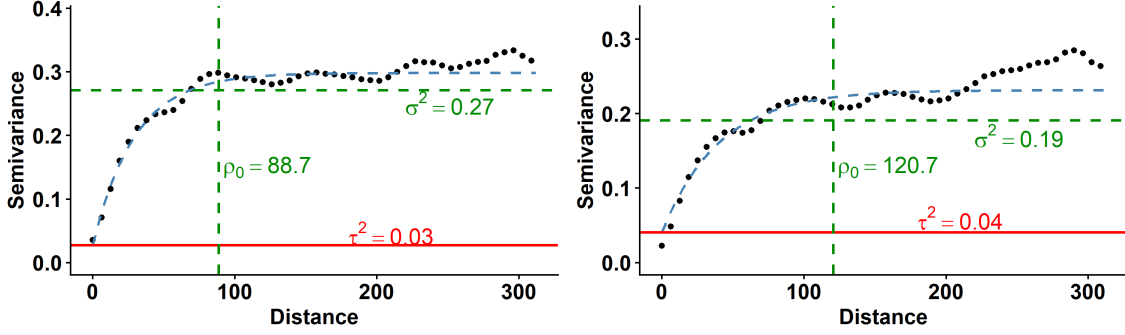


Figure 22: From left to right: sample variograms of NDVI, and Red Reflectance.

of 0.27, and a practical range of approximately 88 based upon automated weighted least squares. This corresponds to significant spatial correlation up to about 10,000 kilometers. The proportion of spatial variability is computed as $\alpha = \sigma^2 / (\tau^2 + \sigma^2) = 0.27 / (0.03 + 0.27) \approx 0.9$, resulting in 0.909 without rounding. Finally, the spatial range parameter is estimated as $\phi = 0.067$ based upon the distance beyond which the spatial correlation drops to less than 0.05; see the left panel of Figure 22.

For RR, the variogram parameters include a nugget effect of 0.04, a sill of 0.19, and a practical range of approximately 120, which corresponds to around 13,000 kilometers. The slightly higher nugget effect for RR suggests greater measurement error or micro-scale variability compared to NDVI. The proportion of spatial variability is estimated as $\alpha = 0.19 / (0.04 + 0.19) \approx 0.825$. The practical range for RR is more extended than that of NDVI, indicating that RR values remain spatially correlated over a greater distance. Concluding the exploratory spatial data analysis, we select a spatial range of $\phi = 0.049$ for RR.

The combined analysis of the variograms for NDVI and RR provides essential information about the spatial variance proportion and range parameters, which are critical for informing artificially intelligent geospatial modeling systems. This analysis results in $\alpha \in \{0.825, 0.909\}$ and $\phi \in \{0.049, 0.067\}$. These findings help improve the accuracy of spatial predictions, enhancing ecological interpretations and increasing computational efficiency by avoiding excessively misspecified model specifications.

The entire exploratory analysis workflow, designed to gather critical insights for improving the DOUBLE BPS methodology, is fully automated. Human input is then minimized; the only required user input is the number of grid values for each spatial parameter.

Using the same subsample composed of 31,875 locations, we also investigated the cross-variogram; see Figure 23. This will help us gain insights into spatial cross-dependencies.

The empirical cross-variogram depicts negative values, providing an estimate for the sill of -0.23 , and a practical range of approximately 85 based upon automated weighted least squares. Similarly to individual variogram analysis, this shows a significant (negative) spatial correlation that withstands up to several thousand kilometers, suggesting a clear and well-defined negative spatial correlation structure between NDVI and red reflectance. The negative cross-variogram mirrors the negative correlation found using the non-spatial model (2). The presence of strong (negative) spatial correlation among these indices is

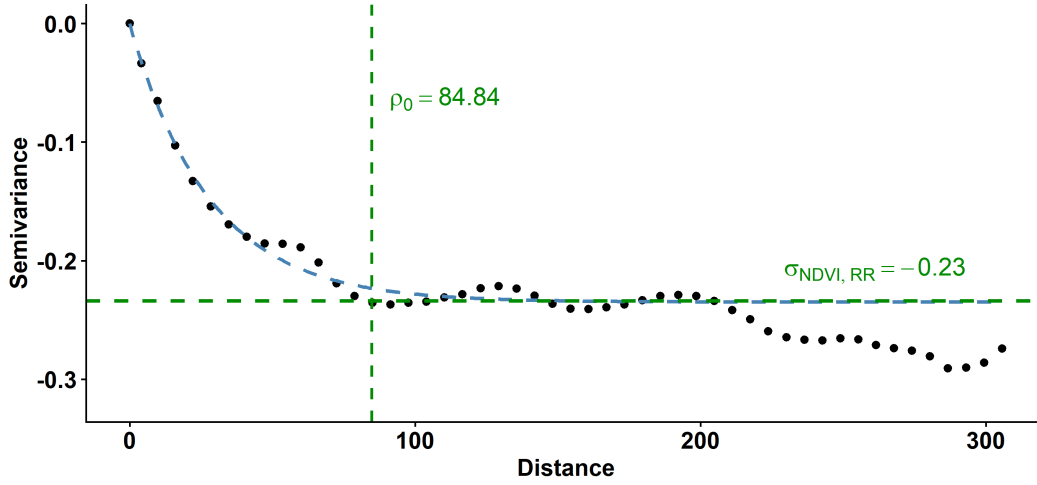


Figure 23: Sample cross-variogram between NDVI, and Red Reflectance.

not surprising, as their definitions are strictly related, and both are based on spectral reflectance measurements acquired in the visible and near-infrared regions. Intuitively, the negative correlation emerged considering that healthy vegetation, which reflects high levels of biomass (NDVI), has strong chlorophyll absorption abilities, then revealing low red reflectance. Conversely, an increase in red reflectance corresponds to stressed (or low) vegetation, which results in low levels of the normalized difference vegetation index. The non-spatial and spatial negative associations are fully consistent with the nature of these indices and with the literature (Tucker, 1979; Sellers, 1985).

References

- J. J. Allaire, Y. Tang, T. Kalinowski, D. Falbel, D. Eddelbuettel, and N. Golding. tensorflow: R Interface to 'TensorFlow', Apr. 2024. URL <https://cran.r-project.org/web/packages/tensorflow/index.html>.
- S. Banerjee. High-dimensional Bayesian geostatistics. *Bayesian Analysis*, 12(2):583–614, 2017. doi: 10.1214/17-BA1056R.
- S. Banerjee. Modeling massive spatial datasets using a conjugate Bayesian linear modeling framework. *Spatial Statistics*, 37:100417, 2020. ISSN 2211-6753. doi: <https://doi.org/10.1016/j.spasta.2020.100417>. URL <https://www.sciencedirect.com/science/article/pii/S2211675320300117>.
- S. Banerjee. Discussion of “Measuring housing vitality from multi-source big data and machine learning”. *Journal of the American Statistical Association*, 117(539):1063–1065, 2022. doi: 10.1080/01621459.2022.2098135.
- S. Banerjee, A. E. Gelfand, A. O. Finley, and H. Sang. Gaussian predictive process models for large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical*

- Methodology*), 70(4):825–848, 2008. doi: <https://doi.org/10.1111/j.1467-9868.2008.00663.x>.
- S. Banerjee, X. Chen, I. Frankenburg, and D. Zhou. Dynamic Bayesian learning for spatiotemporal mechanistic models. *Journal of Machine Learning Research*, 26(146):1–43, 2025a. ISSN 1533-7928. URL <http://jmlr.org/papers/v26/22-0896.html>.
- S. Banerjee, A. E. Gelfand, and B. P. Carlin. *Hierarchical Modeling and Analysis for Spatial Data, Third Edition*. CRC Press, 2025b.
- M. L. Bell, F. Dominici, and J. M. Samet. A meta-analysis of time-series studies of ozone and mortality with comparison to the national morbidity, mortality, and air pollution study. *Epidemiology*, 16(4):436–45, 2005. ISSN 1044-3983. URL https://journals.lww.com/epidem/fulltext/2005/07000/a_meta_analysis_of_time_series_studies_of_ozone.4.aspx.
- D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: a review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017. doi: 10.1080/01621459.2017.1285773.
- L. Breiman. Stacked regressions. *Machine Learning*, 24(1):49–64, 1996. doi: 10.1007/BF00117832.
- D. Cabel, S. Sugawara, M. Kato, K. Takanashi, and K. McAlinn. Bayesian spatial predictive synthesis, Jan. 2025. URL <http://arxiv.org/abs/2203.05197>. arXiv:2203.05197 [stat].
- J. Cao, M. Kang, F. Jimenez, H. Sang, F. T. Schaefer, and M. Katzfuss. Variational sparse inverse cholesky approximation for latent gaussian processes via double kullback-leibler minimization. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 3559–3576. PMLR, July 2023. URL <https://proceedings.mlr.press/v202/cao23b.html>.
- B. Clarke. Comparing bayes model averaging and stacking when model approximation error cannot be ignored. *Journal of Machine Learning Research*, 4(Dec):683–712, 2003. URL <https://www.jmlr.org/papers/v4/clarke03a.html>.
- M. A. Clyde and E. S. Iversen. Bayesian model averaging in the m-open framework. In *Bayesian Theory and Applications*, pages 483–498. Oxford University Press, 01 2013. ISBN 9780199695607. doi: 10.1093/acprof:oso/9780199695607.003.0024.
- N. Cressie and G. Johannesson. Fixed rank kriging for very large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):209–226, 2008. doi: <https://doi.org/10.1111/j.1467-9868.2007.00633.x>. URL <https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2007.00633.x>.
- N. A. C. Cressie. *Statistics for Spatial Data*. Wiley Series in Probability and Statistics. Wiley, 1 edition, Sept. 1993. ISBN 978-0-471-00255-0 978-1-119-11515-1. doi: 10.1002/9781119115151. URL <https://onlinelibrary.wiley.com/doi/book/10.1002/9781119115151>.

- N. A. C. Cressie and C. K. Wikle. *Statistics for spatio-temporal data*. Wiley series in probability and statistics. Wiley, Hoboken, N.J, 2011. ISBN 978-0-471-69274-4 978-1-119-24306-9.
- I. CVX Research. *CVX: Matlab Software for Disciplined Convex Programming, version 2.0*, Aug. 2012.
- A. Datta, S. Banerjee, A. O. Finley, and A. E. Gelfand. Hierarchical nearest-neighbor Gaussian process models for large geostatistical datasets. *Journal of the American Statistical Association*, 111(514):800–812, 2016. doi: 10.1080/01621459.2015.1044091.
- M. Deisenroth and J. W. Ng. Distributed Gaussian processes. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 1481–1490. PMLR, June 2015. URL <https://proceedings.mlr.press/v37/deisenroth15.html>. ISSN: 1938-7228.
- D. Dey, A. Datta, and S. Banerjee. Graphical Gaussian process models for highly multivariate spatial data. *Biometrika*, 109(4):993–1014, Dec. 2022. ISSN 1464-3510. doi: 10.1093/biomet/asab061. URL <https://doi.org/10.1093/biomet/asab061>.
- K. Didan. *MODIS/Terra Vegetation Indices 16-Day L3 Global 0.05Deg CMG V061 | NASA Earthdata*, June 2021. URL <https://www.earthdata.nasa.gov/data/catalog/lpcloud-mod13c1-061>.
- J. Esser, M. Maia, A. C. Parnell, J. Bosmans, H. v. Dongen, T. Klausch, and K. Murphy. Seemingly unrelated Bayesian additive regression trees for cost-effectiveness analyses in healthcare, Feb. 2025. URL <http://arxiv.org/abs/2404.02228>. arXiv:2404.02228 [stat].
- M. Fasiolo. *An introduction to mvnfast.*, 2014. URL <https://CRAN.R-project.org/package=mvnfast>.
- A. O. Finley, S. Banerjee, and A. E. Gelfand. spBayes for large univariate and multivariate point-referenced spatio-temporal data models. *Journal of Statistical Software*, 63(13): 1–28, 2015. doi: 10.18637/jss.v063.i13. URL <https://www.jstatsoft.org/index.php/jss/article/view/v063i13>.
- A. O. Finley, A. Datta, B. D. Cook, D. C. Morton, H.-E. Andersen, and S. Banerjee. Efficient algorithms for Bayesian nearest neighbor Gaussian processes. *Journal of Computational and Graphical Statistics*, 28(2):401–414, 2019.
- A. O. Finley, S. Banerjee, Øyvind Hjelle, and R. Bivand. *MBA: Multilevel B-Spline Approximation*, 2026. URL <https://CRAN.R-project.org/package=MBA>. R package version 0.1-3.
- E. L. Fisher. Hurricanes and the sea-surface temperature field. *Journal of Atmospheric Sciences*, 15(3):328–333, 1958. doi: 10.1175/1520-0469(1958)015<0328:HATSST>2.0.CO;2. URL https://journals.ametsoc.org/view/journals/atsc/15/3/1520-0469_1958_015_0328_hatsst_2_0_co_2.xml.

- D. C.-L. Fong and M. Saunders. LSMR: An iterative algorithm for sparse least-squares problems. *SIAM Journal on Scientific Computing*, 33(5):2950–2971, 2011.
- C. A. Friehe, W. J. Shaw, D. P. Rogers, K. L. Davidson, W. G. Large, S. A. Stage, G. H. Crescenti, S. J. S. Khalsa, G. K. Greenhut, and F. Li. Air-sea fluxes and surface layer turbulence around a sea surface temperature front. *Journal of Geophysical Research: Oceans*, 96(C5):8593–8609, 1991. doi: <https://doi.org/10.1029/90JC02062>. URL <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/90JC02062>.
- T. Fryda, E. LeDell, N. Gill, S. Aiello, A. Fu, A. Candel, C. Click, T. Kraljevic, T. Nykodym, P. Aboyoun, M. Kurka, M. Malohlava, S. Poirier, W. Wong, L. Rehak, E. Eckstrand, B. Hill, S. Vidrio, S. Jadhawani, A. Wang, R. Peck, J. Gorecki, M. Dowle, Y. Tang, L. DiPerna, V. Maurerova, Y. Syzon, A. Valenta, M. Novotny, and H2O.ai. *h2o: R Interface for the 'H2O' Scalable Machine Learning Platform*, Jan. 2024. URL <https://cran.r-project.org/web/packages/h2o/index.html>.
- A. Fu, B. Narasimhan, and S. Boyd. CVXR: An R package for disciplined convex optimization. *Journal of Statistical Software*, 94(14):1–34, 2020. doi: 10.18637/jss.v094.i14.
- A. Fu, B. Narasimhan, F. Schwendinger, and M. Maechler. *ECOSolveR: Embedded Conic Solver in R*, May 2023. URL <https://cran.r-project.org/web/packages/ECOSolveR/index.html>.
- A. Ganguly, S. Jain, and U. Watchareeruetai. Amortized variational inference: a systematic review. *Journal of Artificial Intelligence Research*, 78:167–215, Oct. 2023. ISSN 1076-9757. doi: 10.1613/jair.1.14258. URL <http://dx.doi.org/10.1613/jair.1.14258>.
- A. E. Gelfand and S. Banerjee. Bayesian modeling and analysis of geo-statistical data. *Annual Review of Statistics and Its Application*, 4(Volume 4, 2017):245–266, 2017. ISSN 2326-831X. doi: <https://doi.org/10.1146/annurev-statistics-060116-054155>. URL <https://www.annualreviews.org/content/journals/10.1146/annurev-statistics-060116-054155>.
- A. E. Gelfand, P. Diggle, P. Guttorp, and M. Fuentes. *Handbook of Spatial Statistics*. Taylor & Francis, 2010.
- S. Georganos, T. Grippa, A. N. Gadiaga, C. Linard, M. Lennert, S. Vanhuysse, N. Mboga, E. Wolff, and S. Kalogirou. Geographical random forests: a spatial extension of the random forest algorithm to address spatial heterogeneity in remote sensing and population modelling. *Geocarto International*, 36(2):121–136, 2021. doi: 10.1080/10106049.2019.1595177.
- T. Gneiting and A. E. Raftery. Strictly Proper Scoring Rules, Prediction, and Estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi: 10.1198/016214506000001437.
- M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008.

- M. C. Grant. *Disciplined convex programming*. PhD Thesis, Stanford University, 2005.
- R. Guhaniyogi and S. Banerjee. Meta-Kriging: Scalable Bayesian modeling and inference for massive spatial datasets. *Technometrics*, 60(4):430–444, 2018. doi: 10.1080/00401706.2018.1437474.
- R. Guhaniyogi and S. Banerjee. Multivariate spatial meta kriging. *Statistics & Probability Letters*, 144:3–8, May 2019. ISSN 0167-7152. doi: <https://doi.org/10.1016/j.spl.2018.04.017>. URL <https://www.sciencedirect.com/science/article/pii/S0167715218301718>.
- R. Guhaniyogi, C. Li, T. D. Savitsky, and S. Srivastava. A divide-and-conquer Bayesian approach to large-scale kriging. *arXiv preprint*, page arXiv:1712.09767 [stat.ME], 2019. doi: 10.48550/arXiv.1712.09767.
- R. Guhaniyogi, C. Li, T. D. Savitsky, and S. Srivastava. Distributed Bayesian varying coefficient modeling using a Gaussian process prior. *Journal of Machine Learning Research*, 23(84):1–59, 2022. URL <http://jmlr.org/papers/v23/20-543.html>.
- R. Guhaniyogi, C. Li, T. Savitsky, and S. Srivastava. Distributed Bayesian inference in massive spatial data. *Statistical Science*, 38(2):262–284, May 2023. ISSN 0883-4237, 2168-8745. doi: 10.1214/22-STS868. URL <https://projecteuclid.org/journals/statistical-science/volume-38/issue-2/Distributed-Bayesian-Inference-in-Massive-Spatial-Data/10.1214/22-STS868.full>.
- A. K. Gupta and D. K. Nagar. *Matrix variate distributions*. Monographs and surveys in pure and applied mathematics. Chapman & Hall/CRC, Boca Raton, 2000. ISBN 1-58488-046-5. doi: 10.1201/978-0-2037-4928-9.
- M. A. Haque, M. N. Reza, M. Ali, M. R. Karim, S. Ahmed, K.-D. Lee, Y. H. Khang, and S.-O. Chung. Effects of environmental conditions on vegetation indices from multispectral images: a review. *Korean Journal of Remote Sensing*, 40(4):319–341, 2024. ISSN 1225-6161. doi: 10.7780/kjrs.2024.40.4.1. URL <https://koreascience.or.kr/article/JAK0202424843417603.page>.
- K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition, Dec. 2015. URL <http://arxiv.org/abs/1512.03385>. arXiv:1512.03385 [cs].
- M. J. Heaton, A. Datta, A. O. Finley, R. Furrer, J. Guinness, R. Guhaniyogi, F. Gerber, R. B. Gramacy, D. M. Hammerling, M. Katzfuss, F. Lindgren, D. W. Nychka, F. Sun, and A. Zammit-Mangion. A case study competition among methods for analyzing large spatial data. *Journal of Agricultural, Biological, and Environmental Statistics*, 24(3): 398–425, 2017.
- A. Iranmanesh, M. Arashi, and S. M. M. a. Tabatabaey. On conditional applications of matrix variate Normal distribution. *Iranian Journal of Mathematical Sciences and Informatics*, 5(2):33–43, 2010. doi: 10.7508/ijmsi.2010.02.004. URL <http://ijmsi.ir/article-1-139-en.html>.

- C. O. Justice, J. R. G. Townshend, B. N. Holben, and C. J. Tucker. Analysis of the phenology of global vegetation using meteorological satellite data. *International Journal of Remote Sensing*, 6(8):1271–1318, Aug. 1985. ISSN 0143-1161. doi: 10.1080/01431168508948281. URL <https://doi.org/10.1080/01431168508948281>.
- T. Kalinowski, D. Falbel, J. J. Allaire, F. Chollet, RStudio, Google, Y. Tang [ctb, cph, W. V. D. Bijl, M. Studer, and S. Keydana. keras: R Interface to 'Keras', Apr. 2024. URL <https://cran.r-project.org/web/packages/keras/index.html>.
- J. Kang, T. D. Johnson, T. E. Nichols, and T. D. Wager. Meta Analysis of Functional Neuroimaging Data via Bayesian Spatial Point Processes. *Journal of the American Statistical Association*, 106(493):124–134, 2011. doi: 10.1198/jasa.2011.ap09735.
- M. Katzfuss. A multi-resolution approximation for massive spatial datasets. *Journal of the American Statistical Association*, 112(517):201–214, 2017. doi: 10.1080/01621459.2015.1123632.
- M. Katzfuss and J. Guinness. A general framework for Vecchia approximations of Gaussian processes. *Statistical Science*, 36(1):124–141, 2021. doi: 10.1214/19-STS755. URL <https://doi.org/10.1214/19-STS755>.
- M. Knüppel and F. Krüger. Forecast uncertainty, disagreement, and the linear pool. *Journal of Applied Econometrics*, 37(1):23–41, 2022. ISSN 1099-1255. doi: 10.1002/jae.2834. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/jae.2834>.
- T. Le and B. Clarke. A bayes interpretation of stacking for $\mathcal{M}$ -complete and $\mathcal{M}$ -open settings. *Bayesian Analysis*, 12(3):807–829, 2017. doi: 10.1214/16-BA1023.
- F. Lindgren, H. Rue, and J. Lindström. An explicit link between Gaussian fields and Gaussian Markov random fields: The stochastic partial differential equation approach. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 73(4):423–498, Aug. 2011.
- K. McAlinn and M. West. Dynamic Bayesian predictive synthesis in time series forecasting. *Journal of Econometrics*, 210(1):155–169, May 2019. ISSN 0304-4076. doi: 10.1016/j.jeconom.2018.11.010. URL <https://www.sciencedirect.com/science/article/pii/S0304407618302112>.
- K. McAlinn, A. , Knut Are, N. , Jouchi, , and M. West. Multivariate Bayesian predictive synthesis in macroeconomic forecasting. *Journal of the American Statistical Association*, 115(531):1092–1110, July 2020. ISSN 0162-1459. doi: 10.1080/01621459.2019.1660171. URL <https://doi.org/10.1080/01621459.2019.1660171>.
- Microsoft and S. Weston. *Foreach: Provides foreach looping construct*. Microsoft, 2022. URL <https://CRAN.R-project.org/package=foreach>.
- C. Microsoft and S. Weston. *doParallel: Foreach parallel adaptor for the 'parallel' Package*. Microsoft, 2022. URL <https://CRAN.R-project.org/package=doParallel>.

- S. Minsker, S. Srivastava, L. Lin, and D. B. Dunson. Robust and scalable Bayes via a median of subset posterior measures. *Journal of Machine Learning Research*, 18(124):1–40, 2017.
- N. Nicholls. Sea surface temperatures and australian winter rainfall. *Journal of Climate*, 2(9):965–973, 1989. doi: 10.1175/1520-0442(1989)002<0965:SSTAAW>2.0.CO;2. URL https://journals.ametsoc.org/view/journals/clim/2/9/1520-0442_1989_002_0965_sstaaw_2_0_co_2.xml.
- D. Nychka, S. Bandyopadhyay, D. Hammerling, F. Lindgren, and S. Sain. A multiresolution Gaussian process model for the analysis of large spatial datasets. *Journal of Computational and Graphical Statistics*, 24(2):579–599, 2015. doi: 10.1080/10618600.2014.914946. URL <http://dx.doi.org/10.1080/10618600.2014.914946>.
- A. G. O’Carroll, E. M. Armstrong, H. M. Beggs, M. Bouali, K. S. Casey, G. K. Corlett, P. Dash, C. J. Donlon, C. L. Gentemann, J. L. Hoyer, A. Ignatov, K. Kabobah, M. Kachi, Y. Kurihara, I. Karagali, E. Maturi, C. J. Merchant, S. Marullo, P. J. Minnett, M. Pennybacker, B. Ramakrishnan, R. Ramsankaran, R. Santoleri, S. Sunder, S. Saux Picart, J. Vázquez-Cuervo, and W. Wimmer. Observational needs of sea surface temperature. *Frontiers in Marine Science*, 6(420):1–27, 2019. ISSN 2296-7745. doi: 10.3389/fmars.2019.00420. URL <https://www.frontiersin.org/articles/10.3389/fmars.2019.00420>.
- B. O’Donoghue, E. Chu, N. Parikh, and S. Boyd. Conic optimization via operator splitting and homogeneous self-dual embedding. *Journal of Optimization Theory and Applications*, 169(3):1042–1068, June 2016. ISSN 1573-2878. doi: 10.1007/s10957-016-0892-3. URL <https://doi.org/10.1007/s10957-016-0892-3>.
- S. Pan, L. Zhang, J. R. Bradley, and S. Banerjee. Bayesian inference for spatial-temporal non-gaussian data using predictive stacking. *Bayesian Analysis*, pages 1 – 27, 2025. doi: 10.1214/25-BA1582. URL <https://doi.org/10.1214/25-BA1582>.
- G. Parmigiani. *Modeling in Medical Decision Making: A Bayesian Approach*. John Wiley & Sons, Hoboken, NJ, 2002.
- M. Peruzzi, S. Banerjee, and A. Finley. Highly scalable Bayesian geostatistical modeling via meshed Gaussian processes on partitioned domains. *Journal of the American Statistical Association*, 117(538):969–982, 2022.
- J. Quiñonero-Candela and C. E. Rasmussen. A unifying view of sparse approximate gaussian process regression. *Journal of Machine Learning Research*, 6(65):1939–1959, 2005. URL <http://jmlr.org/papers/v6/quinonero-candela05a.html>.
- Q. Ren, S. Banerjee, A. O. Finley, and J. S. Hodges. Variational Bayesian methods for spatial data analysis. *Computational Statistics & Data Analysis*, 55(12):3197–3217, 2011. ISSN 0167-9473. doi: <https://doi.org/10.1016/j.csda.2011.05.021>. URL <https://www.sciencedirect.com/science/article/pii/S0167947311002003>.

- L. J. Rendell, A. L. Adam M. Johansen, and N. Whiteley. Global consensus Monte Carlo. *Journal of Computational and Graphical Statistics*, 30(2):249–259, 2021. doi: 10.1080/10618600.2020.1811105.
- C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer New York, New York, NY, second edition, 2004. ISBN 978-1-4757-4145-2. doi: 10.1007/978-1-4757-4145-2.
- H. Rue, S. Martino, and N. Chopin. Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 71(2):319–392, Apr. 2009.
- A. Saha, S. Basu, and A. Datta. Random forests for spatially dependent data. *Journal of the American Statistical Association*, 118(541):665–683, 2023. doi: 10.1080/01621459.2021.1950003.
- M. Sainsbury-Dale, A. Zammit-Mangion, J. Richards, and R. Huser. Neural Bayes estimators for irregular spatial data using graph neural networks, 2024. URL <https://arxiv.org/abs/2310.02600>. arXiv:2310.02600.
- A. Sauer, A. Cooper, and R. B. Gramacy. Vecchia-approximated deep Gaussian processes for computer experiments. *Journal of Computational and Graphical Statistics*, 32(3):824–837, 2023a. doi: 10.1080/10618600.2022.2129662.
- A. Sauer, R. B. Gramacy, and D. Higdon. Active learning for deep Gaussian process surrogates. *Technometrics*, 65(1):4–18, 2023b. doi: 10.1080/00401706.2021.2008505.
- S. L. Scott, F. V. B. Alexander W. Blocker, H. A. Chipman, E. I. George, and R. E. McCulloch. Bayes and big data: the consensus Monte Carlo algorithm. *International Journal of Management Science and Engineering Management*, 11(2):78–88, 2016. doi: 10.1080/17509653.2016.1142191.
- P. J. Sellers. Canopy reflectance, photosynthesis and transpiration. *International Journal of Remote Sensing*, 6(8):1335–1372, Aug. 1985. ISSN 0143-1161. doi: 10.1080/01431168508948283. URL <https://doi.org/10.1080/01431168508948283>.
- S. Srivastava, V. Cevher, Q. Dinh, and D. Dunson. WASP: Scalable Bayes via barycenters of subset posteriors. In G. Lebanon and S. V. N. Vishwanathan, editors, *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, pages 912–920, San Diego, California, USA, May 2015. PMLR.
- S. Srivastava, C. Li, and D. B. Dunson. Scalable Bayes via barycenter in Wasserstein space. *Journal of Machine Learning Research*, 19(8):1–35, 2018. URL <http://jmlr.org/papers/v19/17-084.html>.
- M. L. Stein. Asymptotically efficient prediction of a random field with a misspecified covariance function. *The Annals of Statistics*, 16(1):55–63, 1988.

- M. L. Stein. *Interpolation of spatial data*. Springer Series in Statistics. Springer-Verlag, New York, 1999. ISBN 0-387-98629-4. doi: 10.1007/978-1-4612-1494-6.
- M. L. Stein and M. S. Handcock. Some asymptotic properties of kriging when the covariance function is misspecified. *Mathematical Geology*, 21:171–190, 1989. doi: <https://doi.org/10.1007/BF00893213>.
- P. M. Suder, J. Xu, and D. B. Dunson. Bayesian transfer learning. *Statistical Science*, 40(3): 477 – 494, 2025. doi: 10.1214/25-STS987. URL <https://doi.org/10.1214/25-STS987>.
- H. Talebi, L. J. M. Peeters, A. Otto, and R. Tolosana-Delgado. A truly spatial random forests algorithm for geoscience data analysis and modelling. *Mathematical Geosciences*, 54(1):1–22, Jan. 2022. ISSN 1874-8953. doi: 10.1007/s11004-021-09946-w. URL <https://doi.org/10.1007/s11004-021-09946-w>.
- E. Tallman and M. West. Bayesian predictive decision synthesis. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(2):340–363, Oct. 2023. ISSN 1369-7412. doi: 10.1093/jrsssb/qkad109. URL <https://doi.org/10.1093/jrsssb/qkad109>.
- W. Tang, L. Zhang, and S. Banerjee. On identifiability and consistency of the nugget in Gaussian spatial process models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 83(5):1044–1070, 2021. doi: <https://doi.org/10.1111/rssb.12472>. URL <https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/rssb.12472>.
- C. J. Tucker. Red and photographic infrared linear combinations for monitoring vegetation. *Remote Sensing of Environment*, 8(2):127–150, May 1979. ISSN 0034-4257. doi: 10.1016/0034-4257(79)90013-0. URL <https://www.sciencedirect.com/science/article/pii/0034425779900130>.
- A. V. Vecchia. Estimation and model identification for continuous spatial processes. *Journal of the Royal Statistical society, Series B*, 50:297–312, 1988.
- C. K. Wikle. Low-rank representations for spatial processes. *Handbook of Spatial Statistics*, pages 107–118, 2010.
- C. K. Wikle and A. Zammit-Mangion. Statistical deep learning for spatial and spatiotemporal data. *Annual Review of Statistics and Its Application*, 10 (Volume 10, 2023):247–270, 2023. ISSN 2326-831X. doi: <https://doi.org/10.1146/annurev-statistics-033021-112628>. URL <https://www.annualreviews.org/content/journals/10.1146/annurev-statistics-033021-112628>.
- D. H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992. doi: [https://doi.org/10.1016/S0893-6080\(05\)80023-1](https://doi.org/10.1016/S0893-6080(05)80023-1).
- L. Wu, G. Pleiss, and J. P. Cunningham. Variational nearest neighbor Gaussian process. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 24114–24130. PMLR, July 2022. URL <https://proceedings.mlr.press/v162/wu22h.html>.

- Y. Yao, A. Vehtari, D. Simpson, and A. Gelman. Using stacking to average bayesian predictive distributions (with discussion). *Bayesian Analysis*, 13(3):917–1007, 2018. doi: 10.1214/17-BA1091.
- Y. Yao, G. Pirs, A. Vehtari, and A. Gelman. Bayesian hierarchical stacking: Some models are (somewhere) useful. *Bayesian Analysis*, 17(4):1043 – 1071, 2022. doi: 10.1214/21-BA1287. URL <https://doi.org/10.1214/21-BA1287>.
- A. Zammit-Mangion, T. L. J. Ng, Q. Vu, and M. Filippone. Deep compositional spatial models. *Journal of the American Statistical Association*, 117(540):1787–1808, 2022. doi: 10.1080/01621459.2021.1887741.
- A. Zammit-Mangion, M. Sainsbury-Dale, and R. Huser. Neural methods for amortized inference. *Annual Review of Statistics and Its Application*, 12(Volume 12, 2025):311–335, 2025. ISSN 2326-831X. doi: <https://doi.org/10.1146/annurev-statistics-112723-034123>. URL <https://www.annualreviews.org/content/journals/10.1146/annurev-statistics-112723-034123>.
- H. Zhang. Inconsistent estimation and asymptotically equal interpolations in model-based geostatistics. *Journal of the American Statistical Association*, 99(465):250–261, 2004.
- L. Zhang and S. Banerjee. Spatial factor modeling: A Bayesian matrix-normal approach for misaligned data. *Biometrics*, 78(2):560–573, 2022. doi: <https://doi.org/10.1111/biom.13452>.
- L. Zhang, A. Datta, and S. Banerjee. Practical Bayesian modeling and inference for massive spatial data sets on modest computing environments. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 12(3):197–209, 2019. doi: <https://doi.org/10.1002/sam.11413>.
- L. Zhang, S. Banerjee, and A. O. Finley. High-dimensional multivariate geostatistics: A Bayesian matrix-normal approach. *Environmetrics*, 32(4):e2675, 2021. doi: <https://doi.org/10.1002/env.2675>.
- L. Zhang, W. Tang, and S. Banerjee. Bayesian geostatistics using predictive stacking. *Journal of the American Statistical Association*, (in press):1–19, 2025. doi: 10.1080/01621459.2025.2566449.
- Y. Zhou, L. Xue, Z. Shi, L. Wu, and J. Fan. Measuring housing vitality from multi-source big data and machine learning. *Journal of the American Statistical Association*, 117(539): 1045–1059, 2022. doi: 10.1080/01621459.2022.2096038.