

Supplement: Anisotropic Multidimensional Smoothing Using Bayesian Tensor Product P-Splines

Paul Bach and Nadja Klein

This document is structured as follows:

- Appendix [A](#) derives the full conditional posteriors of β and σ^2 and explains how the regression coefficients β can be updated efficiently using sparse Cholesky updates.
- Appendix [B](#) contains proofs for our theoretical results of Section 3 in the paper.
- Appendix [C](#) contains proofs for our theoretical results of Section 4 in the paper.
- Appendix [D](#) explains how a prior scaling approach can be used to determine a reasonable value for the Weibull rate parameter.
- Appendix [E](#) states the partial derivatives of the log-kernel when the smoothing variances follow inverse gamma or Weibull priors.
- Appendix [F](#) provides additional information for the temperature data set analyzed in Section 5.2 of the paper.
- Appendix [G](#) contains another real data example about two-dimensional spatial smoothing of US precipitation anomalies.

A Efficient MCMC Updates for the Regression Coefficients

In this appendix we derive the full conditional posteriors of β and σ^2 and explain how the regression coefficients β can be updated efficiently using sparse Cholesky updates.

A.1 Full Conditional Posteriors

The full conditional posterior of β is proportional to

$$\begin{aligned} p(\beta \mid \cdot) &\propto p(\mathbf{y} \mid \beta, \sigma^2) p(\beta \mid \tau^2) \\ &\propto N_n(\mathbf{y}; \mathbf{B}\beta, \sigma^2 \mathbf{I}_n) \exp\left(-\frac{1}{2}\beta^\top \mathbf{K}(\tau^2)\beta\right) \\ &\propto \exp\left(-\frac{\|\mathbf{y} - \mathbf{B}\beta\|_2^2}{2\sigma^2}\right) \exp\left(-\frac{1}{2}\beta^\top \mathbf{K}(\tau^2)\beta\right) \\ &\propto \exp\left(-\frac{1}{2}\left\{\beta^\top \left[\frac{\mathbf{B}^\top \mathbf{B}}{\sigma^2} + \mathbf{K}(\tau^2)\right]\beta - \beta^\top \frac{\mathbf{B}^\top \mathbf{y}}{\sigma^2}\right\}\right), \quad \beta \in \mathbb{R}^D, \end{aligned}$$

where we recognize the kernel of a D -variate Gaussian density with covariance matrix $\Sigma = [\mathbf{B}^\top \mathbf{B}/\sigma^2 + \mathbf{K}(\tau^2)]^{-1}$ and mean vector $\mu = [\mathbf{B}^\top \mathbf{B}/\sigma^2 + \mathbf{K}(\tau^2)]^{-1} \mathbf{B}^\top \mathbf{y}/\sigma^2$. Thus, the full conditional posterior distribution of β is

$$\beta \mid \cdot \sim N_D([\mathbf{B}^\top \mathbf{B}/\sigma^2 + \mathbf{K}(\tau^2)]^{-1} \mathbf{B}^\top \mathbf{y}/\sigma^2, [\mathbf{B}^\top \mathbf{B}/\sigma^2 + \mathbf{K}(\tau^2)]^{-1}).$$

The full conditional posterior of σ^2 is proportional to

$$\begin{aligned} p(\sigma^2 \mid \cdot) &\propto p(\mathbf{y} \mid \beta, \sigma^2) p(\sigma^2) \\ &\propto N_n(\mathbf{y}; \mathbf{B}\beta, \sigma^2 \mathbf{I}_n) 1/\sigma^2 \\ &\propto (\sigma^2)^{-n/2-1} \exp(-\|\mathbf{y} - \mathbf{B}\beta\|_2^2/(2\sigma^2)), \quad \sigma^2 > 0, \end{aligned}$$

where we recognize the kernel of an inverse gamma density with shape $n/2$ and scale $\|\mathbf{y} - \mathbf{B}\beta\|_2^2/2$. Thus, the full conditional posterior distribution of σ^2 is

$$\sigma^2 \mid \cdot \sim IG(n/2, \|\mathbf{y} - \mathbf{B}\beta\|_2^2/2).$$

A.2 Sparse Cholesky Updates

The MCMC update of β has the form

$$\beta \mid \cdot \sim N_D(\mathbf{Q}^{-1}\mathbf{v}, \mathbf{Q}^{-1}), \tag{1}$$

where $\mathbf{Q}$ is a symmetric positive definite precision matrix and $\mathbf{v} \in \mathbb{R}^D$ is a vector. Let $\mathbf{L}$ be a Cholesky factor of the precision matrix such that $\mathbf{Q} = \mathbf{L}\mathbf{L}^\top$. Then the inverse of $\mathbf{Q}$ is equal to $\mathbf{Q}^{-1} = (\mathbf{L}\mathbf{L}^\top)^{-1} = (\mathbf{L}^\top)^{-1}\mathbf{L}^{-1}$ and it holds

$$\begin{aligned} N_D(\mathbf{Q}^{-1}\mathbf{v}, \mathbf{Q}^{-1}) &= N_D((\mathbf{L}^\top)^{-1}\mathbf{L}^{-1}\mathbf{v}, (\mathbf{L}^\top)^{-1}\mathbf{L}^{-1}) \\ &\stackrel{d}{=} (\mathbf{L}^\top)^{-1}N_D(\mathbf{L}^{-1}\mathbf{v}, \mathbf{I}_D) \\ &\stackrel{d}{=} (\mathbf{L}^\top)^{-1}(N_D(\mathbf{0}, \mathbf{I}_D) + \mathbf{L}^{-1}\mathbf{v}), \end{aligned}$$

where $\stackrel{d}{=}$ means equal in distribution. Thus, the coefficient vector $\boldsymbol{\beta}$ can be updated efficiently using the following steps:

1. Compute the Cholesky decomposition $\mathbf{Q} = \mathbf{L}\mathbf{L}^\top$ of the precision matrix $\mathbf{Q}$;
2. Solve $\mathbf{L}\mathbf{w} = \mathbf{v}$ for $\mathbf{w}$;
3. Sample $\mathbf{z} \sim N_D(\mathbf{0}, \mathbf{I}_D)$;
4. Solve $\mathbf{L}^\top\boldsymbol{\beta} = \mathbf{z} + \mathbf{w}$ for $\boldsymbol{\beta}$.

Then $\boldsymbol{\beta}$ follows the desired distribution (1). This procedure is a slightly simplified version of Algorithm 2.5 in [Rue and Held \(2005\)](#) (we only need to solve two triangular systems of equations, while they need to solve three). The corresponding R code is:

```
R=Matrix::chol(Q); L=t(R)
Matrix::solve(R, rnorm(n=D)+Matrix::solve(L, v))
```

The R package `Matrix` ([Bates et al., 2022](#)) allows us to exploit the sparsity of the precision matrix $\mathbf{Q}$ for the Cholesky decomposition and to make use of the triangular structure when solving the two systems of equations.

A.3 Blockwise Updates

The update of $\boldsymbol{\beta}$ becomes time consuming for very large dimension (say $D > 1000$). In this case, one can partition the vector $\boldsymbol{\beta}$ into several blocks to reduce the computational burden (see, e.g., [Ishwaran and Rao, 2005](#), Appendix). For simplicity, we explain the approach for two blocks. To this end, we partition the design matrix $\mathbf{B} = (\mathbf{B}_1, \mathbf{B}_2)$ and the coefficient vector $\boldsymbol{\beta} = (\boldsymbol{\beta}_1^\top, \boldsymbol{\beta}_2^\top)^\top$ as well as the overall roughness penalty matrix $\mathbf{K} = \mathbf{K}(\tau^2)$ into blocks of matching sizes

$$\mathbf{K} = \begin{pmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{pmatrix}.$$

Then, the update of $\boldsymbol{\beta}_1$ is $N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ with

$$\boldsymbol{\Sigma}_1 = \left(\frac{\mathbf{B}_1^\top \mathbf{B}_1}{\sigma^2} + \mathbf{K}_{11} \right)^{-1} \quad \text{and} \quad \boldsymbol{\mu}_1 = \boldsymbol{\Sigma}_1 \left(\frac{\mathbf{B}_1^\top \tilde{\mathbf{y}}}{\sigma^2} - \mathbf{K}_{12}\boldsymbol{\beta}_2 \right),$$

where $\tilde{\mathbf{y}} = \mathbf{y} - \mathbf{B}_2\boldsymbol{\beta}_2$. Moreover, the update of $\boldsymbol{\beta}_2$ is $N(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ with

$$\boldsymbol{\Sigma}_2 = \left(\frac{\mathbf{B}_2^\top \mathbf{B}_2}{\sigma^2} + \mathbf{K}_{22} \right)^{-1} \quad \text{and} \quad \boldsymbol{\mu}_2 = \boldsymbol{\Sigma}_2 \left(\frac{\mathbf{B}_2^\top \tilde{\mathbf{y}}}{\sigma^2} - \mathbf{K}_{21}\boldsymbol{\beta}_1 \right),$$

where $\tilde{\mathbf{y}} = \mathbf{y} - \mathbf{B}_1\boldsymbol{\beta}_1$. In general (for any number of blocks) the update of $\boldsymbol{\beta}_{j_0}$ is $N(\boldsymbol{\mu}_{j_0}, \boldsymbol{\Sigma}_{j_0})$ with

$$\boldsymbol{\Sigma}_{j_0} = \left(\frac{\mathbf{B}_{j_0}^\top \mathbf{B}_{j_0}}{\sigma^2} + \mathbf{K}_{j_0, j_0} \right)^{-1} \quad \text{and} \quad \boldsymbol{\mu}_{j_0} = \boldsymbol{\Sigma}_{j_0} \left(\frac{\mathbf{B}_{j_0}^\top \tilde{\mathbf{y}}}{\sigma^2} - \sum_{j \neq j_0} \mathbf{K}_{j_0, j} \boldsymbol{\beta}_j \right),$$

where $\tilde{\mathbf{y}} = \mathbf{y} - \sum_{j \neq j_0} \mathbf{B}_j \boldsymbol{\beta}_j$. As the updates again have the form (1), we can use the approach of the previous section for efficient sampling.

B Proofs for Section 3

In this appendix we prove the theoretical results of Section 3 in the paper.

Proof of Theorem 3.1. To prove the result for arbitrary dimension $p \in \mathbb{N}$, it will be convenient to first introduce the Kronecker sum of two square matrices. Let $\mathbf{A}$ be a matrix of size $m \times m$ and $\mathbf{B}$ be a matrix of size $n \times n$. Following [Graham \(2018\)](#) we define the Kronecker sum of $\mathbf{A}$ and $\mathbf{B}$ as the $mn \times mn$ matrix

$$\mathbf{A} \oplus \mathbf{B} = \mathbf{A} \otimes \mathbf{I}_n + \mathbf{I}_m \otimes \mathbf{B}.$$

The summation extends naturally to more than two summands and can easily be shown to be associative. With this, we can write the overall roughness penalty matrix $\mathbf{K}(\boldsymbol{\tau}^2)$ as the Kronecker sum of the marginal penalty matrices, i.e.

$$\mathbf{K}(\boldsymbol{\tau}^2) = \frac{\widetilde{\mathbf{K}}_1}{\tau_1^2} \oplus \cdots \oplus \frac{\widetilde{\mathbf{K}}_p}{\tau_p^2}.$$

Moreover, we can write the statement which we aim to prove in the equivalent form

$$\frac{\widetilde{\mathbf{K}}_1}{\tau_1^2} \oplus \cdots \oplus \frac{\widetilde{\mathbf{K}}_p}{\tau_p^2} = \left(\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p \right) \left(\frac{\tilde{\boldsymbol{\Gamma}}_1}{\tau_1^2} \oplus \cdots \oplus \frac{\tilde{\boldsymbol{\Gamma}}_p}{\tau_p^2} \right) \left(\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p \right)^\top.$$

Since $\mathbf{Q} = \tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p$ is orthogonal, it suffices to show that

$$\left(\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p \right)^\top \left(\frac{\widetilde{\mathbf{K}}_1}{\tau_1^2} \oplus \cdots \oplus \frac{\widetilde{\mathbf{K}}_p}{\tau_p^2} \right) \left(\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p \right) = \frac{\tilde{\boldsymbol{\Gamma}}_1}{\tau_1^2} \oplus \cdots \oplus \frac{\tilde{\boldsymbol{\Gamma}}_p}{\tau_p^2}. \quad (2)$$

We prove statement (2) by induction over p . For $p = 1$ the statement is obviously satisfied since $\tilde{\mathbf{Q}}_1^\top \widetilde{\mathbf{K}}_1 \tilde{\mathbf{Q}}_1 / \tau_1^2 = \tilde{\boldsymbol{\Gamma}}_1 / \tau_1^2$. We assume that the statement is satisfied for p and show that it is also satisfied for $p + 1$. It holds:

$$\begin{aligned}
& \left(\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_{p+1} \right)^\top \left(\frac{\tilde{\mathbf{K}}_1}{\tau_1^2} \oplus \cdots \oplus \frac{\tilde{\mathbf{K}}_{p+1}}{\tau_{p+1}^2} \right) \left(\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_{p+1} \right) \\
&= \left((\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p) \otimes \tilde{\mathbf{Q}}_{p+1} \right)^\top \left(\left(\frac{\tilde{\mathbf{K}}_1}{\tau_1^2} \oplus \cdots \oplus \frac{\tilde{\mathbf{K}}_p}{\tau_p^2} \right) \oplus \frac{\tilde{\mathbf{K}}_{p+1}}{\tau_{p+1}^2} \right) \\
&\quad \left((\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p) \otimes \tilde{\mathbf{Q}}_{p+1} \right) \\
&\stackrel{(2)}{=} \left((\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p)^\top \otimes \tilde{\mathbf{Q}}_{p+1}^\top \right) \left(\left(\frac{\tilde{\mathbf{K}}_1}{\tau_1^2} \oplus \cdots \oplus \frac{\tilde{\mathbf{K}}_p}{\tau_p^2} \right) \otimes \mathbf{I}_{d_{p+1}} + \mathbf{I}_{d_1 \dots d_p} \otimes \frac{\tilde{\mathbf{K}}_{p+1}}{\tau_{p+1}^2} \right) \\
&\quad \left((\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p) \otimes \tilde{\mathbf{Q}}_{p+1} \right) \\
&\stackrel{(3)}{=} \left((\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p)^\top \left(\frac{\tilde{\mathbf{K}}_1}{\tau_1^2} \oplus \cdots \oplus \frac{\tilde{\mathbf{K}}_p}{\tau_p^2} \right) \otimes \tilde{\mathbf{Q}}_{p+1}^\top + (\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p)^\top \otimes \frac{\tilde{\mathbf{Q}}_{p+1}^\top \tilde{\mathbf{K}}_{p+1}}{\tau_{p+1}^2} \right) \\
&\quad \left((\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p) \otimes \tilde{\mathbf{Q}}_{p+1} \right) \\
&\stackrel{(4)}{=} (\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p)^\top \left(\frac{\tilde{\mathbf{K}}_1}{\tau_1^2} \oplus \cdots \oplus \frac{\tilde{\mathbf{K}}_p}{\tau_p^2} \right) (\tilde{\mathbf{Q}}_1 \otimes \cdots \otimes \tilde{\mathbf{Q}}_p) \otimes \mathbf{I}_{d_{p+1}} + \mathbf{I}_{d_1 \dots d_p} \otimes \frac{\tilde{\mathbf{Q}}_{p+1}^\top \tilde{\mathbf{K}}_{p+1} \tilde{\mathbf{Q}}_{p+1}}{\tau_{p+1}^2} \\
&\stackrel{(5)}{=} \left(\frac{\tilde{\mathbf{\Gamma}}_1}{\tau_1^2} \oplus \cdots \oplus \frac{\tilde{\mathbf{\Gamma}}_p}{\tau_p^2} \right) \otimes \mathbf{I}_{d_{p+1}} + \mathbf{I}_{d_1 \dots d_p} \otimes \frac{\tilde{\mathbf{\Gamma}}_{p+1}}{\tau_{p+1}^2} \\
&\stackrel{(6)}{=} \left(\frac{\tilde{\mathbf{\Gamma}}_1}{\tau_1^2} \oplus \cdots \oplus \frac{\tilde{\mathbf{\Gamma}}_p}{\tau_p^2} \right) \oplus \frac{\tilde{\mathbf{\Gamma}}_{p+1}}{\tau_{p+1}^2} = \frac{\tilde{\mathbf{\Gamma}}_1}{\tau_1^2} \oplus \cdots \oplus \frac{\tilde{\mathbf{\Gamma}}_{p+1}}{\tau_{p+1}^2},
\end{aligned}$$

where we have used the definition of the Kronecker sum for the second equality, the properties of the Kronecker product (distributivity and the mixed-product property) for the third and fourth equality, the induction hypothesis for the fifth equality and the definition of the Kronecker sum for the sixth equality. This finishes the proof of Theorem 3.1. $\square$

Proof of Corollary 3.2. The first equality follows from Theorem 3.1 and the fact that similar matrices have the same eigenvalues. The second equality follows from the diagonality of the $\mathbf{\Gamma}_j$, $j = 1, \dots, p$, and the fact that the pseudo-determinant of a diagonal matrix is simply the product of the nonzero diagonal entries. $\square$

C Proofs for Section 4

In this appendix we prove the theoretical results of Section 4 in the paper.

Proof of Proposition 4.1. Let $\mathbf{x}_{-j} = (x_1, \dots, x_{j-1}, x_{j+1}, \dots, x_p)^\top \in [0, 1]^{p-1}$ be arbitrary but fixed. Then it holds:

$$\begin{aligned}
f(\mathbf{x}) &= \mathbf{B}(\mathbf{x})^\top \boldsymbol{\beta} \\
&= (\vec{\mathbf{B}}_1(x_1)^\top \otimes \dots \otimes \vec{\mathbf{B}}_j(x_j)^\top \otimes \dots \otimes \vec{\mathbf{B}}_p(x_p)^\top) \boldsymbol{\beta} \\
&= (1 \vec{\mathbf{B}}_1(x_1)^\top \otimes \dots \otimes \vec{\mathbf{B}}_j(x_j)^\top \mathbf{I}_{d_j} \otimes \dots \otimes 1 \vec{\mathbf{B}}_p(x_p)^\top) \boldsymbol{\beta} \\
&= (1 \otimes \dots \otimes \vec{\mathbf{B}}_j(x_j)^\top \otimes \dots \otimes 1) (\vec{\mathbf{B}}_1(x_1)^\top \otimes \dots \otimes \mathbf{I}_{d_j} \otimes \dots \otimes \vec{\mathbf{B}}_p(x_p)^\top) \boldsymbol{\beta} \\
&= \vec{\mathbf{B}}_j(x_j)^\top (\vec{\mathbf{B}}_1(x_1)^\top \otimes \dots \otimes \mathbf{I}_{d_j} \otimes \dots \otimes \vec{\mathbf{B}}_p(x_p)^\top) \boldsymbol{\beta}, \quad x_j \in [0, 1],
\end{aligned}$$

where we used the mixed-product property of the Kronecker product for the fourth equality and the fact that $1 \otimes M = M \otimes 1 = M$ for an arbitrary matrix M for the last equality. This finishes the proof for one-dimensional slices. The proof for two-dimensional slices follows by the same arguments and is therefore omitted. $\square$

Proof of Proposition 4.2. It holds:

$$\begin{aligned}
\int_{[0,1]^{p-1}} f(\mathbf{x}) d\mathbf{x}_{-j} &= \int_{[0,1]^{p-1}} \vec{\mathbf{B}}(\mathbf{x})^\top \boldsymbol{\beta} d\mathbf{x}_{-j} \\
&= \left(\int_0^1 \vec{\mathbf{B}}_1(x_1)^\top dx_1 \otimes \dots \otimes \vec{\mathbf{B}}_j(x_j)^\top \otimes \dots \otimes \int_0^1 \vec{\mathbf{B}}_p(x_p)^\top dx_p \right) \boldsymbol{\beta} \\
&= (\mathbf{A}_1^\top \otimes \dots \otimes \vec{\mathbf{B}}_j(x_j)^\top \otimes \dots \otimes \mathbf{A}_p^\top) \boldsymbol{\beta}, \quad x_j \in [0, 1],
\end{aligned}$$

where we used Fubini's theorem and linearity of the integral for the second equality and the definition of the $\mathbf{A}_j$ for the third equality. Using the same arguments as in the proof of Proposition 4.1, we obtain that the previous display is equal to

$$\begin{aligned}
& (1 \mathbf{A}_1^\top \otimes \dots \otimes \vec{\mathbf{B}}_j(x_j)^\top \mathbf{I}_{d_j} \otimes \dots \otimes 1 \mathbf{A}_p^\top) \boldsymbol{\beta} \\
&= (1 \otimes \dots \otimes \vec{\mathbf{B}}_j(x_j)^\top \otimes \dots \otimes 1) (\mathbf{A}_1^\top \otimes \dots \otimes \mathbf{I}_{d_j} \otimes \dots \otimes \mathbf{A}_p^\top) \boldsymbol{\beta} \\
&= \vec{\mathbf{B}}_j(x_j)^\top (\mathbf{A}_1^\top \otimes \dots \otimes \mathbf{I}_{d_j} \otimes \dots \otimes \mathbf{A}_p^\top) \boldsymbol{\beta}, \quad x_j \in [0, 1],
\end{aligned}$$

which finishes the proof for one-dimensional averages. The proof for two-dimensional averages follows by the same arguments and is therefore omitted. $\square$

D Prior Scaling

In this appendix we explain how a prior scaling approach (cf. [Klein and Kneib, 2016](#)) can be used to determine a reasonable value for a common Weibull rate parameter λ . The key idea of the prior scaling approach is to set λ such that prior function draws f have a reasonable scale. More formally, we aim to set λ such that

$$\mathbb{P}(\|f\| \geq c) \stackrel{!}{=} \alpha,$$

where $\|\cdot\|$ is a suitable norm, $c > 0$ is a threshold and $\alpha \in (0, 1)$ is a probability level. All three quantities are chosen by the user and we state our defaults at the end of this section.

The partial impropriety of the conditional prior of $\boldsymbol{\beta}$ precludes direct sampling from the prior of f . As a remedy, we exploit the mixed model representation of tensor product smooths using $\boldsymbol{Q} = (\boldsymbol{Q}_+, \boldsymbol{Q}_0)$ from Theorem 3.1 as change-of-basis matrix. This allows us to decompose the vector of function evaluations $\boldsymbol{f}^n = (f(\boldsymbol{x}_1), \dots, f(\boldsymbol{x}_n))^\top$ into a penalized part and an unpenalized part, i.e., we can write $\boldsymbol{f}^n$ in the form

$$\boldsymbol{f}^n = \boldsymbol{B}\boldsymbol{\beta} = \boldsymbol{Z}\boldsymbol{u} + \boldsymbol{X}\boldsymbol{\zeta},$$

where $p(\boldsymbol{\zeta}) \propto 1$ is improper uniform and $\boldsymbol{u} \mid \boldsymbol{\tau}^2 \sim N(\mathbf{0}, \boldsymbol{\Gamma}_+(\boldsymbol{\tau}^2)^{-1})$. Thereby, $\boldsymbol{\Gamma}_+(\boldsymbol{\tau}^2)$ is the diagonal matrix corresponding to the positive diagonal entries of $\boldsymbol{\Gamma}(\boldsymbol{\tau}^2) = \boldsymbol{\Gamma}_1/\tau_1^2 + \dots + \boldsymbol{\Gamma}_p/\tau_p^2$. Moreover, $\boldsymbol{Z} = \boldsymbol{B}\boldsymbol{Q}_+$ and $\boldsymbol{X} = \boldsymbol{B}\boldsymbol{Q}_0$. With this, we proceed as follows:

a) Draw a large Monte-Carlo sample from

$$\|\boldsymbol{Z}\boldsymbol{u}\|_\infty \quad \text{with} \quad \boldsymbol{u} \mid \boldsymbol{\tau}^2 \sim N(\mathbf{0}, \boldsymbol{\Gamma}_+(\boldsymbol{\tau}^2)^{-1}) \quad \text{and} \quad \tau_j^2 \stackrel{iid}{\sim} Weibull(1/2, 1), \quad j = 1, \dots, p.$$

b) Determine the empirical $(1 - \alpha)$ quantile $q_{1-\alpha}$ of the sample from a).

c) Set $\lambda = (q_{1-\alpha}/c)^2$.

This procedure ensures that

$$\mathbb{P}(\|\boldsymbol{Z}\boldsymbol{u}\|_\infty \geq c) = \alpha$$

up to Monte-Carlo error. By default, we use the absolute norm of the observations for the threshold c , i.e., we set $c = \|\boldsymbol{y}\|_\infty$. In addition to that we use $\alpha = 0.05$ for the probability level and generate 1,000 Monte-Carlo samples by default.

E Partial Derivatives of the Log-Kernel

To implement the Taylored MH updates for the inverse gamma prior and the Weibull prior we need the log-kernel in terms of $\boldsymbol{\rho}$, i.e. $\log q(\boldsymbol{\rho})$, as well as its first and second order partial derivatives.

E.1 Inverse Gamma

By the density transformation formula, the joint prior of $\boldsymbol{\rho}$ is proportional to

$$p(\boldsymbol{\rho}) \propto \prod_{j=1}^p IG(e^{\rho_j}; a_j, b_j) \prod_{j=1}^p e^{\rho_j} \propto \prod_{j=1}^p (e^{\rho_j})^{-a_j} \exp(-b_j e^{-\rho_j}), \quad \boldsymbol{\rho} \in \mathbb{R}^p.$$

Thus, the log-kernel is

$$\log q(\boldsymbol{\rho}) = \sum_{j=1}^p -a_j \rho_j - b_j e^{-\rho_j}, \quad \boldsymbol{\rho} \in \mathbb{R}^p.$$

For the partial derivatives we thus obtain

$$\begin{aligned} \partial_{j_0} \log q(\boldsymbol{\rho}) &= -a_{j_0} + b_{j_0} e^{-\rho_{j_0}}, \\ \partial_{j_0}^2 \log q(\boldsymbol{\rho}) &= -b_{j_0} e^{-\rho_{j_0}}, \\ \partial_{j_0} \partial_{k_0} \log q(\boldsymbol{\rho}) &= 0, \quad \text{if } j_0 \neq k_0. \end{aligned}$$

E.2 Weibull with Shape 1/2

By the density transformation formula, the joint prior of $\boldsymbol{\rho}$ is proportional to

$$p(\boldsymbol{\rho}) \propto \prod_{j=1}^p Weibull(e^{\rho_j}; 1/2, \lambda_j) \prod_{j=1}^p e^{\rho_j} \propto \prod_{j=1}^p e^{\rho_j/2} \exp(-\sqrt{\lambda_j} e^{\rho_j/2}), \quad \boldsymbol{\rho} \in \mathbb{R}^p.$$

Thus, the log-kernel is

$$\log q(\boldsymbol{\rho}) = - \sum_{j=1}^p \sqrt{\lambda_j} e^{\rho_j/2} + \frac{1}{2} \sum_{j=1}^p \rho_j, \quad \boldsymbol{\rho} \in \mathbb{R}^p.$$

For the partial derivatives we thus obtain

$$\begin{aligned} \partial_{j_0} \log q(\boldsymbol{\rho}) &= -\frac{1}{2} \sqrt{\lambda_{j_0}} e^{\rho_{j_0}/2} + \frac{1}{2}, \\ \partial_{j_0}^2 \log q(\boldsymbol{\rho}) &= -\frac{1}{4} \sqrt{\lambda_{j_0}} e^{\rho_{j_0}/2}, \\ \partial_{j_0} \partial_{k_0} \log q(\boldsymbol{\rho}) &= 0, \quad \text{if } j_0 \neq k_0. \end{aligned}$$

F Additional Information for the Temperature Data

In this appendix we provide additional background information for the temperature data set analyzed in Section 5.2 of the paper. Moreover, we explain our preprocessing steps.

F.1 Berkeley Earth Climate Data Set

We have downloaded the Berkeley Earth climate data set from kaggle (www.kaggle.com) using the following link:

www.kaggle.com/datasets/berkeleyearth/climate-change-earth-surface-temperature-data

The data set has been provided by Berkely Earth (www.berkeleyearth.org) and is licensed under CC BY-NC-SA 4.0 allowing for non-commercial use. There are several csv-files (together about 600 MB) containing global land and ocean temperature data with records dating back as early as 1750.

F.2 Preprocessing

We have selected the csv-file `GlobalLandTemperaturesByCity.csv` and conducted the following preprocessing steps to create the temperature data set analyzed in Section 5.2 of the paper.

- First we have selected the records from the USA.
- Then we have selected the records from 2000 up to 2010.
- Next we have removed the records for the city Anchorage because it is too far away from the other measurement locations (spatial outlier).
- Next we have removed duplicates (the file `GlobalLandTemperaturesByCity.csv` contains several identical time series that are associated with the same geographical location but have different city labels, e.g. the temperature time series for San Francisco and Oakland are identical).
- Finally, we have converted longitude and latitude to numbers to facilitate the analysis.

The resulting data set contains temperature time series of length $11 \times 12 = 132$ for 96 different locations across the USA. In total we thus have $n = 132 \times 96 = 12,672$ records of the monthly average temperature.

G US precipitation anomalies

For this example we closely follow Section 6.1 of [Rodríguez-Álvarez et al. \(2015\)](#) and analyze US precipitation anomalies for April 1948. The analyzed data set `USprecip` is available in the R package `spam`, further details about the data are provided in [Rodríguez-Álvarez et al. \(2015\)](#). Following these authors we restrict our analysis to those $n = 5,906$ observations that have a valid precipitation record. Figure 1 panel A visualizes the raw data.

We apply our new method with Weibull hyperpriors for estimation of a smooth precipitation anomaly surface. Following [Rodríguez-Álvarez et al. \(2015\)](#) we use 44-dimensional marginal B-spline bases leading to $D = 44 \times 44 = 1,936$ tensor product B-splines. Such a high number of basis functions seems to be necessary to capture the complexity of the surface. The estimated surface is shown in Figure 1 panel B and is virtually indistinguishable from that of [Rodríguez-Álvarez et al. \(2015\)](#). Panels C and D show lower and upper simultaneous 95% credible surfaces, respectively. To this end, we used the method of [Krivobokova et al. \(2010\)](#). Estimation takes only 3 minutes for 2,500 MCMC samples from which we discard the first 500 as burn-in. The median effective sample size (ESS) across the β_j is 1,935 (with minimum 1,044) indicating good MCMC mixing ([Vehtari et al. \(2021\)](#) recommend an ESS of at least 400 for each parameter of interest).

For comparison, we apply the `SAP` algorithm of [Rodríguez-Álvarez et al. \(2015\)](#). With this, we need about 5 minutes for estimation of the anomaly surface. This is remarkable as `SAP` is based on REML, which is usually much faster than MCMC, but at the cost of more involved and only asymptotically valid uncertainty quantification (see, e.g., [Wood, 2020](#)). A possible explanation for the comparable speed is that [Rodríguez-Álvarez et al. \(2015\)](#) use a mixed model reparametrization, which leads to non-sparse matrices in their algorithm. For a larger number of basis functions, our approach even becomes much faster than `SAP`: For 80-dimensional marginal B-spline bases (leading to $D = 80 \times 80 = 6,400$ tensor product B-splines) we need about 11 minutes for 2,500 MCMC samples (median/min ESS 1943/772), whereas the `SAP` algorithm needs about 180 minutes to compute an estimate.

This example shows that our approach can compete and in certain settings even outperform the `SAP` algorithm. In addition, our approach provides straightforward uncertainty quantification, which does not even seem to be implemented for the `SAP` algorithm of [Rodríguez-Álvarez et al. \(2015\)](#).

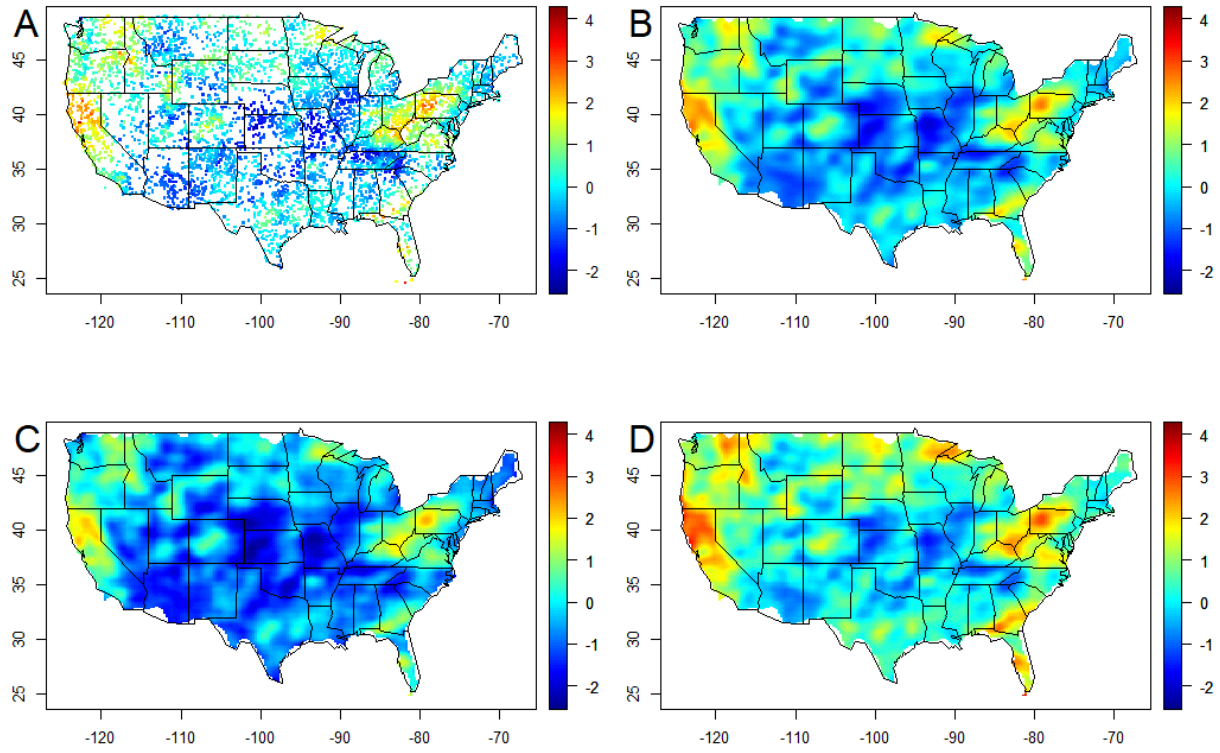


Figure 1: US precipitation anomalies. Shown are the raw data (panel A), the estimated surface from our approach (panel B) as well as lower and upper 95% credible surfaces (panels C and D, respectively).

References

- Bates, D., Maechler, M., and Jagan, M. (2022). *Matrix: Sparse and Dense Matrix Classes and Methods*. R package version 1.5-1.
- Graham, A. (2018). *Kronecker Products and Matrix Calculus with Applications*. Dover Publications, Mineola, NY.
- Ishwaran, H. and Rao, J. S. (2005). Spike and slab variable selection: Frequentist and Bayesian strategies. *The Annals of Statistics*, 33(2):730–773.
- Klein, N. and Kneib, T. (2016). Scale-dependent priors for variance parameters in structured additive distributional regression. *Bayesian Analysis*, 11(4):1071–1106.
- Krivobokova, T., Kneib, T., and Claeskens, G. (2010). Simultaneous confidence bands for penalized spline estimators. *Journal of the American Statistical Association*, 105(490):852–863.
- Rodríguez-Álvarez, M. X., Lee, D.-J., Kneib, T., Durbán, M., and Eilers, P. (2015). Fast smoothing parameter separation in multidimensional generalized P-splines: The SAP algorithm. *Statistics and Computing*, 25(5):941–957.
- Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory and Applications*. Chapman & Hall/CRC press, Boca Raton, FL.
- Vehtari, A., Gelman, A., Simpson, D., Carpenter, B., and Bürkner, P.-C. (2021). Rank-normalization, folding, and localization: An improved R for assessing convergence of MCMC (with discussion). *Bayesian Analysis*, 16(2):667–718.
- Wood, S. N. (2020). Inference and computation with generalized additive models and their extensions. *TEST*, pages 1–33.