Web-based Supplementary Materials for *"Marginal Bayesian Nonparametric Model for Time to Disease Arrival of Threatened Amphibian Populations"* by Haiming Zhou, Timothy Hanson, and Roland Knapp

1 Web Appendix A: MCMC for the Marginal LDDPM Spatial Survival Model

The full likelihood function for the data is given by

$$
\begin{aligned}
\mathcal{L}(\mathbf{y},\mathbf{B},\boldsymbol{\sigma},\mathbf{K},\mathbf{V},\alpha,\boldsymbol{\mu},\boldsymbol{\theta}) \propto & \prod_{i=1}^{n}(\sigma_{K_i}^2)^{-1/2}\exp\left\{-\frac{1}{2\sigma_{K_i}^2}(y_i-\mathbf{x}_i'\boldsymbol{\beta}_{K_i})^2\right\} \\
& \times\left\{\delta_i I(y_i=y_i^o)+(1-\delta_i)I(y_i>y_i^o)\right\} \\
& \times|\mathbf{C}_{\boldsymbol{\theta}}|^{-1/2}\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1}-\mathbf{I}_n)\mathbf{z}\right\} \\
& \times\prod_{i=1}^{n}\left[V_{K_i}\prod_{k<K_i}(1-V_k)\right] \\
& \times\left\{\prod_{k=1}^{N-1}\frac{\Gamma(\alpha+1)}{\Gamma(\alpha)}(1-V_k)^{\alpha-1}\right\}\times\alpha^{a_0-1}\exp\{-b_0\alpha\} \\
& \times\prod_{k=1}^{N}|\boldsymbol{\Sigma}|^{-1/2}\exp\left\{-\frac{1}{2}(\boldsymbol{\beta}_k-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{\beta}_k-\boldsymbol{\mu})\right\} \\
& \times\prod_{k=1}^{N}\left(\sigma_k^{-2}\right)^{\nu_a-1}\exp\left\{-\nu_b\sigma_k^{-2}\right\} \\
& \times|\mathbf{S}_0|^{-1/2}\exp\left\{-\frac{1}{2}(\boldsymbol{\mu}-\mathbf{m}_0)'\mathbf{S}_0^{-1}(\boldsymbol{\mu}-\mathbf{m}_0)\right\} \\
& \times|\boldsymbol{\Sigma}|^{-(\kappa_0-p-1)/2}\exp\left\{-\frac{1}{2}\mathrm{tr}\left(\kappa_0\boldsymbol{\Sigma}_0\boldsymbol{\Sigma}^{-1}\right)\right\} \\
& \times\theta_1^{\theta_{1a}-1}(1-\theta_1)^{\theta_{1b}-1}\times\theta_2^{\theta_{2a}-1}e^{-\theta_{2b}\theta_2}
\end{aligned}
\tag{A.1}
$$

Step 1: Update K_i for $i = 1, \ldots, n$.

The full conditional distribution for K_i is

$$f(K_i|\text{else}) \propto \omega_{K_i}(2\pi\sigma^2_{K_i})^{-1/2}\exp\left\{-\frac{1}{2\sigma^2_{K_i}}(y_i - \mathbf{x}_i'\boldsymbol{\beta}_{K_i})^2\right\}$$
$$\propto \omega_{K_i}\phi(y_i|\mathbf{x}_i'\boldsymbol{\beta}_{K_i}, \sigma^2_{K_i}).$$

It follows that

$$P(K_i = k) = \frac{\omega_k\phi(y_i|\mathbf{x}_i'\boldsymbol{\beta}_k, \sigma^2_k)}{\sum_{k=1}^N \omega_k\phi(y_i|\mathbf{x}_i'\boldsymbol{\beta}_k, \sigma^2_k)}.$$

Step 2: Update y_i for $i = 1, \ldots, n$.
The full conditional distribution for y_i is

$$f(y_i|\text{else}) \propto \delta_i I(y_i = y_i^o) + (1-\delta_i)\phi(y_i|\mathbf{x}_i'\boldsymbol{\beta}_{K_i}, \sigma^2_{K_i})\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}\right\} I(y_i > y_i^o).$$

If $\delta_i = 1$, update $y_i = y_i^o$. If $\delta_i = 0$, propose y_i^* from $N(\mathbf{x}_i'\boldsymbol{\beta}_{K_i}, \sigma^2_{K_i})$ distribution truncated above at y_i^o and accept it with probability

$$\min\left\{1, \frac{\exp\left\{-\frac{1}{2}\mathbf{z}^{*\prime}(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}^*\right\}}{\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}\right\}}\right\},$$

where $\mathbf{z}^* = (z_i^*, \ldots, z_n^*)'$ is the new transformed vector corresponding to y_i^*.

Step 3: Update $\boldsymbol{\beta}_k$ for $k = 1, \ldots, N$ using Metropolis-Hastings algorithms with delayed rejection (Tierney and Mira, 1999).
The full conditional distribution for $\boldsymbol{\beta}_k$ is

$$f(\boldsymbol{\beta}_k|\text{else}) \propto \exp\left\{-\frac{1}{2}(\boldsymbol{\beta}_k - \boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{\beta}_k - \boldsymbol{\mu}) - \sum_{\{i:K_i=k\}}\frac{1}{2\sigma^2_k}(y_i - \mathbf{x}_i'\boldsymbol{\beta}_k)^2\right\}$$
$$\times \exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}\right\}$$
$$\propto \exp\left\{-\frac{1}{2}\boldsymbol{\beta}_k'\left(\boldsymbol{\Sigma}^{-1} + \sigma_k^{-2}\sum_{\{i:K_i=k\}}\mathbf{x}_i\mathbf{x}_i'\right)\boldsymbol{\beta}_k + \boldsymbol{\beta}_k'\left(\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \sigma_k^{-2}\sum_{\{i:K_i=k\}}\mathbf{x}_i y_i\right)\right\}$$
$$\times \exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}\right\}$$
$$\propto \exp\left\{-\frac{1}{2}(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k^*)'(\boldsymbol{\Sigma}_k^*)^{-1}(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k^*)\right\}\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}\right\}$$
$$\propto \phi_p(\boldsymbol{\beta}_k|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}\right\},$$

where

$$\mathbf{\Sigma}_k = \left(\mathbf{\Sigma}^{-1} + \sigma_k^{-2} \sum_{\{i:K_i=k\}} \mathbf{x}_i \mathbf{x}_i' \right)^{-1}$$

$$\boldsymbol{\mu}_k = \mathbf{\Sigma}_k^* \left(\mathbf{\Sigma}^{-1} \boldsymbol{\mu} + \sigma_k^{-2} \sum_{\{i:K_i=k\}} \mathbf{x}_i y_i \right).$$

Propose $\boldsymbol{\beta}_k^*$ from $N_p(\boldsymbol{\mu}_k, \mathbf{\Sigma}_k)$ and accept it with probability

$$\alpha_1(\boldsymbol{\beta}_k, \boldsymbol{\beta}_k^*) = \min\left\{ 1, \frac{\exp\left\{-\frac{1}{2}\mathbf{z}^{*\prime}(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}^*\right\}}{\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}\right\}} \right\}.$$

If $\boldsymbol{\beta}_k^*$ is rejected, we further propose $\boldsymbol{\beta}_k^{**}$ from $N_p(\boldsymbol{\beta}_k, \mathbf{\Sigma}_k)$ and accept it with probability $\alpha_2(\boldsymbol{\beta}_k, \boldsymbol{\beta}_k^*, \boldsymbol{\beta}_k^{**})$ as

$$\begin{aligned}
&\min\left\{ 1, \frac{f(\boldsymbol{\beta}_k^{**}|\text{else})[1 - \alpha_1(\boldsymbol{\beta}_k^{**}, \boldsymbol{\beta}_k^*)]}{f(\boldsymbol{\beta}_k|\text{else})[1 - \alpha_1(\boldsymbol{\beta}_k, \boldsymbol{\beta}_k^*)]} \right\} \\
&= \min\left\{ 1, \frac{\phi_p(\boldsymbol{\beta}_k^{**}|\boldsymbol{\mu}_k, \mathbf{\Sigma}_k)\left[\exp\left\{-\frac{1}{2}\mathbf{z}^{**\prime}(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}^{**} + \frac{1}{2}\mathbf{z}^{*\prime}(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}^*\right\} - 1\right]}{\phi_p(\boldsymbol{\beta}_k|\boldsymbol{\mu}_k, \mathbf{\Sigma}_k)\left[\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z} + \frac{1}{2}\mathbf{z}^{*\prime}(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}^*\right\} - 1\right]} \right\}.
\end{aligned}$$

Step 4: Update σ_k^2 for $k = 1, \ldots, N$.
The full conditional distribution for σ_k^{-2} is

$$\begin{aligned}
f(\sigma_k^{-2}|\text{else}) &\propto \left(\sigma_k^{-2}\right)^{\nu_a - 1} \exp\left\{-\nu_b \sigma_k^{-2}\right\} \prod_{\{i:K_i=k\}} (\sigma_k^2)^{-1/2} \exp\left\{-\frac{1}{2\sigma_k^2}(y_i - \mathbf{x}_i'\boldsymbol{\beta}_k)^2\right\} \\
&\quad \times \exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}\right\} \\
&\propto \left(\sigma_k^{-2}\right)^{\nu_a + n_k/2 - 1} \exp\left\{-\left(\nu_b + \frac{1}{2}\sum_{\{i:K_i=k\}}(y_i - \mathbf{x}_i'\boldsymbol{\beta}_k)^2\right)\sigma_k^{-2}\right\} \\
&\quad \times \exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}\right\} \\
&\propto \text{Ga}\left(\sigma_k^{-2}|\nu_a + n_k/2, \nu_{b,k}^*\right) \exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}\right\},
\end{aligned}$$

where

$$n_k = \sum_{i=1}^n I(K_i = k) \quad \text{and} \quad \nu_{b,k}^* = \nu + \frac{1}{2}\sum_{\{i:K_i=k\}}(y_i - \mathbf{x}_i'\boldsymbol{\beta}_k)^2.$$

Propose $\sigma_k^{-2(*)}$ from $\text{Ga}\left(\nu_a + n_k/2, \nu_{b,k}^*\right)$ and accept it with probability

$$\min\left\{ 1, \frac{\exp\left\{-\frac{1}{2}\mathbf{z}^{*\prime}(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}^*\right\}}{\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1} - \mathbf{I}_n)\mathbf{z}\right\}} \right\}.$$

Step 5: Update V_k for $k = 1, \ldots, N-1$.
Let $n_k = \sum_{i=1}^{n} I(K_i = k)$. Note that $\mathbf{z}$ depends on the values of V_k's through the weithts w_k's. Thus, the full conditional distribution V_i is

$$f(V_k|\text{else}) \propto \prod_{i=1}^{n}\left[V_{K_i}\prod_{k<K_i}(1-V_k)\right] \times \prod_{k=1}^{N-1}(1-V_k)^{\alpha-1}\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1}-\mathbf{I}_n)\mathbf{z}\right\}$$
$$\propto V_k^{n_k}(1-V_k)^{n_{k+1}+n_{k+2}+\ldots+n_N+\alpha-1}\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1}-\mathbf{I}_n)\mathbf{z}\right\}.$$

Propose V_k^* from Beta $\left(1+n_k, \alpha+\sum_{j=k+1}^{N} n_j\right)$ and accept it with probability

$$\min\left\{1, \frac{\exp\left\{-\frac{1}{2}\mathbf{z}^{*\prime}(\mathbf{C}_{\boldsymbol{\theta}}^{-1}-\mathbf{I}_n)\mathbf{z}^*\right\}}{\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}_{\boldsymbol{\theta}}^{-1}-\mathbf{I}_n)\mathbf{z}\right\}}\right\}.$$

Step 6: Update α.
The full conditional distribution for α is

$$f(\alpha|\text{else}) \propto \left\{\prod_{k=1}^{N-1}\frac{\Gamma(\alpha+1)}{\Gamma(\alpha)}(1-V_k)^{\alpha}\right\} \times \alpha^{a_0-1}\exp\{-b_0\alpha\}$$
$$\propto \alpha^{a_0+N-2}\exp\left\{-\alpha\left(b_0-\sum_{k=1}^{N-1}\log(1-V_k)\right)\right\}$$
$$\propto \text{Ga}\left(a_0+N-1, b_0-\sum_{k=1}^{N-1}\log(1-V_k)\right).$$

Step 7: Update $\boldsymbol{\mu}$.
The full conditional distribution for $\boldsymbol{\mu}$ is

$$f(\boldsymbol{\mu}|\text{else}) \propto \exp\left\{-\frac{1}{2}(\boldsymbol{\mu}-\mathbf{m}_0)'\mathbf{S}_0^{-1}(\boldsymbol{\mu}-\mathbf{m}_0)-\frac{1}{2}\sum_{k=1}^{N}(\boldsymbol{\mu}-\boldsymbol{\beta}_k)'\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}-\boldsymbol{\beta}_k)\right\}$$
$$\propto \exp\left\{-\frac{1}{2}\left(\boldsymbol{\mu}-\mathbf{m}_0^*\right)'\left(\mathbf{S}_0^*\right)^{-1}\left(\boldsymbol{\mu}-\mathbf{m}_0^*\right)\right\}$$
$$\propto N_p\left(\mathbf{m}_0^*, \mathbf{S}_0^*\right),$$

where

$$\mathbf{S}_0^* = \left(\mathbf{S}_0^{-1}+N\boldsymbol{\Sigma}^{-1}\right)^{-1} \quad \text{and} \quad \mathbf{m}_0^* = \mathbf{S}_0^*\left(\mathbf{S}_0^{-1}\mathbf{m}_0+\boldsymbol{\Sigma}^{-1}\sum_{k=1}^{N}\boldsymbol{\beta}_k\right).$$

Step 8: Update $\boldsymbol{\Sigma}$.

The full conditional distribution for $\mathbf{\Sigma}^{-1}$ is

$$f(\mathbf{\Sigma}^{-1}|\text{else}) \propto |\mathbf{\Sigma}|^{-(\kappa_0+1-p-1)/2}\exp\left\{-\frac{1}{2}\sum_{k=1}^{N}(\boldsymbol{\mu}-\boldsymbol{\beta}_k)'\mathbf{\Sigma}^{-1}(\boldsymbol{\mu}-\boldsymbol{\beta}_k) - \frac{1}{2}\text{tr}\left(\kappa_0\mathbf{\Sigma}_0\mathbf{\Sigma}^{-1}\right)\right\}$$
$$\propto W_p\left(\left(\kappa_0\mathbf{\Sigma}_0 + \sum_{k=1}^{N}(\boldsymbol{\mu}-\boldsymbol{\beta}_k)(\boldsymbol{\mu}-\boldsymbol{\beta}_k)'\right)^{-1},\ \kappa_0+N\right).$$

Step 9: Update $\boldsymbol{\theta} = (\theta_1, \theta_2)'$ using adaptive Metropolis-Hastings algorithms (Haario et al., 2001).
Let $\boldsymbol{\vartheta} = c(\vartheta_1, \vartheta_2)'$ with $\vartheta_1 = \log\left(\frac{\theta_1}{1-\theta_1}\right)$ and $\vartheta_2 = \log(\theta_2)$. Then, the full conditional distribution for $\boldsymbol{\vartheta}$ is

$$f(\boldsymbol{\vartheta}|\text{else}) \propto |\mathbf{C}_{\boldsymbol{\theta}}|^{-1/2}\exp\left\{-\frac{1}{2}\mathbf{z}'\mathbf{C}_{\boldsymbol{\theta}}^{-1}\mathbf{z}\right\} \times \theta_1^{\theta_{1a}-1}(1-\theta_1)^{\theta_{1b}-1}\theta_1^2 e^{-\vartheta_1} \times \theta_2^{\theta_{2a}-1}e^{-\theta_{2b}\theta_2}e^{\vartheta_2}$$
$$\propto \exp\Big\{-\frac{1}{2}\log|\mathbf{C}_{\boldsymbol{\theta}}| - \frac{1}{2}\mathbf{z}'\mathbf{C}_{\boldsymbol{\theta}}^{-1}\mathbf{z}$$
$$+ (\theta_{1a}+1)\log(\theta_1) + (\theta_{1b}-1)\log(1-\theta_1) - \vartheta_1 + \theta_{2a}\log\theta_2 - \theta_{2b}\theta_2\Big\}.$$

Suppose we are currently in time l and have sampled the states $\boldsymbol{\vartheta}_0, \boldsymbol{\vartheta}_1, \ldots, \boldsymbol{\vartheta}_{l-1}$. We select an index $l_0 > 0$ for the length of an initial period and define

$$S_l = \begin{cases} S_0, & l \le l_0 \\ \frac{(2.4)^2}{2}(\mathcal{C}_l + 0.05 I_2) & l > l_0. \end{cases}$$

Here $\mathcal{C}_l$ is the sample variance of $\boldsymbol{\vartheta}_0, \boldsymbol{\vartheta}_1, \ldots, \boldsymbol{\vartheta}_{l-1}$. Set $\bar{\boldsymbol{\vartheta}}_l = \frac{1}{l}\sum_{i=0}^{l-1}\boldsymbol{\vartheta}_i$. Following Haario et al. (2001), we can use the recursive equations

$$\bar{\boldsymbol{\vartheta}}_{l+1} = \frac{l}{l+1}\bar{\boldsymbol{\vartheta}}_l + \frac{1}{l+1}\boldsymbol{\vartheta}_l$$

and

$$\mathcal{C}_{l+1} = \text{cov}(\boldsymbol{\vartheta}_0, \boldsymbol{\vartheta}_1, \ldots, \boldsymbol{\vartheta}_l) = \frac{1}{l}\sum_{i=0}^{l}\boldsymbol{\vartheta}_i\boldsymbol{\vartheta}_i' - \frac{l+1}{l}\bar{\boldsymbol{\vartheta}}_{l+1}\bar{\boldsymbol{\vartheta}}_{l+1}'$$
$$= \frac{l-1}{l}\mathcal{C}_l + \bar{\boldsymbol{\vartheta}}_l\bar{\boldsymbol{\vartheta}}_l' - \frac{l+1}{l}\bar{\boldsymbol{\vartheta}}_{l+1}\bar{\boldsymbol{\vartheta}}_{l+1}' + \boldsymbol{\vartheta}_l\boldsymbol{\vartheta}_l'/l.$$

It follows that for $l > l_0$

$$S_{l+1} = \frac{l-1}{l}S_l + \frac{(2.4)^2}{2l}\left(l\bar{\boldsymbol{\vartheta}}_l\bar{\boldsymbol{\vartheta}}_l' - (l+1)\bar{\boldsymbol{\vartheta}}_{l+1}\bar{\boldsymbol{\vartheta}}_{l+1}' + \boldsymbol{\vartheta}_l\boldsymbol{\vartheta}_l' + 0.05 I_p\right).$$

We propose $\boldsymbol{\vartheta}^*$ from $N_2(\boldsymbol{\vartheta}_{l-1}, S_l)$ and accept it with probability

$$\min\left\{1, \frac{f(\boldsymbol{\vartheta}^*|\text{else})}{f(\boldsymbol{\vartheta}_{l-1}|\text{else})}\right\},$$

in which case we set $\boldsymbol{\vartheta}_l = \boldsymbol{\vartheta}^*$, and otherwise $\boldsymbol{\vartheta}_l = \boldsymbol{\vartheta}_{l-1}$.

2 Web Appendix B: The Full Scale Approximation of the Spatial Correlation Matrix $\mathbf{C}_{\boldsymbol{\theta}}$

A computational bottleneck of the MCMC sampling scheme is inverting the $n \times n$ matrix $\mathbf{C}_{\boldsymbol{\theta}}$, which typically has computational cost $O(n^3)$. In this section, we introduce a full scale approximation (FSA) approach proposed by Sang and Huang (2012), which provides a high quality approximation to the correlation function ρ at both the large and the small spatial scales, such that the inverse of $\mathbf{C}_{\boldsymbol{\theta}}$ can be substantially sped up for large value of n, e.g., $n \geq 500$.

Consider a fixed set of "knots" $\mathcal{S}^* = \{\mathbf{s}_1^*, \ldots, \mathbf{s}_m^*\}$ chosen from the study region. The FSA approach approximates the correlation function $\rho(\mathbf{s}, \mathbf{s}')$ with

$$\rho^\dagger(\mathbf{s}, \mathbf{s}') = \rho_l(\mathbf{s}, \mathbf{s}') + \rho_s(\mathbf{s}, \mathbf{s}'). \tag{B.2}$$

The $\rho_l(\mathbf{s}, \mathbf{s}')$ in (B.2) is the reduced-rank part capturing the long-scale spatial dependence, defined as $\rho_l(\mathbf{s}, \mathbf{s}') = \rho'(\mathbf{s}, \mathcal{S}^*)\rho_{mm}^{-1}(\mathcal{S}^*, \mathcal{S}^*)\rho(\mathbf{s}', \mathcal{S}^*)$, where $\rho(\mathbf{s}, \mathcal{S}^*) = [\rho(\mathbf{s}, \mathbf{s}_i^*)]_{i=1}^m$ is an $m \times 1$ vector, and $\rho_{mm}(\mathcal{S}^*, \mathcal{S}^*) = [\rho(\mathbf{s}_i^*, \mathbf{s}_j^*)]_{i,j=1}^m$ is an $m \times m$ correlation matrix at knots $\mathcal{S}^*$. However, $\rho_l(\mathbf{s}, \mathbf{s}')$ cannot well capture the short-scale dependence due to the fact that it discards entirely the residual part $\rho(\mathbf{s}, \mathbf{s}') - \rho_l(\mathbf{s}, \mathbf{s}')$. The idea of FSA is to add a small-scale part $\rho_s(\mathbf{s}, \mathbf{s}')$ as a sparse approximate of the residual part, defined by $\rho_s(\mathbf{s}, \mathbf{s}') = \{\rho(\mathbf{s}, \mathbf{s}') - \rho_l(\mathbf{s}, \mathbf{s}')\}\, \Delta(\mathbf{s}, \mathbf{s}')$, where $\Delta(\mathbf{s}, \mathbf{s}')$ is a modulating function, which is specified so that the $\rho_s(\mathbf{s}, \mathbf{s}')$ can well capture the local residual spatial dependence while still permits efficient computations. Motivated by Konomi et al. (2014), we first partition the total input space into B disjoint blocks, and then specify $\Delta(\mathbf{s}, \mathbf{s}')$ in a way such that the residuals are independent across input blocks, but the original residual dependence structure within each block is retained. Specifically, the function $\Delta(\mathbf{s}, \mathbf{s}')$ is taken to be 1 if $\mathbf{s}$ and $\mathbf{s}'$ belong to the same block and 0 otherwise.The approximated correlation function $\rho^\dagger(\mathbf{s}, \mathbf{s}')$ in (B.2) provides an exact recovery of the true correlation within each block, and the approximation errors are $\rho(\mathbf{s}, \mathbf{s}') - \rho_l(\mathbf{s}, \mathbf{s}')$ for locations $\mathbf{s}$ and $\mathbf{s}'$ in different blocks. Those errors are expected to be small for most entries because most of these location pairs are farther apart.

Applying the above FSA approach to approximate the correlation function $\rho(\mathbf{s}, \mathbf{s}')$, we can approximate the correlation matrix $\boldsymbol{\rho}_{nn}$ with

$$\boldsymbol{\rho}_{nn}^\dagger = \boldsymbol{\rho}_l + \boldsymbol{\rho}_s = \boldsymbol{\rho}_{nm}\boldsymbol{\rho}_{mm}^{-1}\boldsymbol{\rho}_{nm}' + \left(\boldsymbol{\rho}_{nn} - \boldsymbol{\rho}_{nm}\boldsymbol{\rho}_{mm}^{-1}\boldsymbol{\rho}_{nm}'\right) \circ \boldsymbol{\Delta}, \tag{B.3}$$

where $\boldsymbol{\rho}_{nm} = [\rho(\mathbf{s}_i, \mathbf{s}_j^*)]_{i=1:n, j=1:m}$, $\boldsymbol{\rho}_{mm} = [\rho(\mathbf{s}_i^*, \mathbf{s}_j^*)]_{i,j=1}^m$, and $\boldsymbol{\Delta} = [\Delta(\mathbf{s}_i, \mathbf{s}_j)]_{i,j=1}^n$. Here, the notation "$\circ$" represents the element-wise matrix multiplication. It follows from equation (B.3) that the covariance matrix $\mathbf{C}_{\boldsymbol{\theta}}$ can be approximated by

$$\mathbf{C}_{\boldsymbol{\theta}}^\dagger = \theta_1\boldsymbol{\rho}_{nn}^\dagger + (1-\theta_1)\mathbf{I}_n = \theta_1\boldsymbol{\rho}_{nm}\boldsymbol{\rho}_{mm}^{-1}\boldsymbol{\rho}_{nm}' + \mathbf{C}_s,$$

where $\mathbf{C}_s = \theta_1\boldsymbol{\rho}_s + (1-\theta_1)\mathbf{I}_n$. Applying the Sherman-Woodbury-Morrison formula for inverse matrices, we can approximate $\mathbf{C}_{\boldsymbol{\theta}}^{-1}$ by $\left(\mathbf{C}_{\boldsymbol{\theta}}^\dagger\right)^{-1}$ which is given by

$$\mathbf{C}_s^{-1} - \theta_1\mathbf{C}_s^{-1}\boldsymbol{\rho}_{nm}\left(\boldsymbol{\rho}_{mm} + \theta_1\boldsymbol{\rho}_{nm}'\mathbf{C}_s^{-1}\boldsymbol{\rho}_{nm}\right)^{-1}\boldsymbol{\rho}_{nm}'\mathbf{C}_s^{-1}. \tag{B.4}$$

In addition, the determinant of $\mathbf{C}_{\boldsymbol{\theta}}$ can be approximated by $\det\left(\mathbf{C}_{\boldsymbol{\theta}}^{\dagger}\right)$ given as

$$\det\left\{\boldsymbol{\rho}_{mm} + \theta_1 \boldsymbol{\rho}_{nm}' \mathbf{C}_s^{-1} \boldsymbol{\rho}_{nm}\right\} \det(\boldsymbol{\rho}_{mm})^{-1} \det(\mathbf{C}_s). \tag{B.5}$$

Since the $n \times n$ matrix $\mathbf{C}_s$ is a block matrix, the right-hand sides of equations (B.4) and (B.5) involve only inverses and determinants of $m \times m$ low-rank matrices and $n \times n$ block diagonal matrices. Thus the computational complexity can be greatly reduced relative to the expensive computational cost of using original correlation function for large value of n.

3 Web Appendix C: Derivation of the CPO Statistic

Let $f_{\mathbf{x}_i}$ and $F_{\mathbf{x}_i}$ be the density and distribution functions of T_i given $\mathbf{x}_i$, respectively. According to the hierarchical representation in Section 2.2, given all the model parameters $\Theta = (\mathbf{B}, \boldsymbol{\sigma}^2, \mathbf{V}, \boldsymbol{\theta})$, we have

$$f_{\mathbf{x}_i}(t_i|\Theta) = \frac{1}{t_i} \sum_{k=1}^{N} w_k \frac{1}{\sigma_k} \phi\left(\frac{\log t_i - \mathbf{x}_i' \boldsymbol{\beta}_k}{\sigma_k}\right),$$
$$F_{\mathbf{x}_i}(t_i|\Theta) = \sum_{k=1}^{N} w_k \frac{1}{\sigma_k} \Phi\left(\frac{\log t_i - \mathbf{x}_i' \boldsymbol{\beta}_k}{\sigma_k}\right),$$

According to the definition of CPO, we have $\text{CPO}_i = f\left(t_i^o|\mathcal{D}_{-i}\right)^{\delta_i} S\left(t_i^o|\mathcal{D}_{-i}\right)^{1-\delta_i}$. Note that, for $\delta_i = 1$,

$$\begin{aligned}
E_{(\mathbf{t},\Theta|\mathcal{D})}\left\{\frac{1}{f(t_i^o|\mathbf{t}_{-i},\Theta)}\right\} &= \frac{\int \frac{1}{f(t_i^o|\mathbf{t}_{-i},\Theta)} f(\mathbf{t}|\Theta)\pi(\Theta)d\Theta \prod_{\{j:\delta_j=0\}} I(t_j > t_j^o)dt_j}{\int f(\mathbf{t}|\Theta)\pi(\Theta)d\Theta \prod_{\{j:\delta_j=0\}} I(t_j > t_j^o)dt_j} \\
&= \frac{\int f(\mathbf{t}_{-i}|\Theta)\pi(\Theta)d\Theta \prod_{\{j:\delta_j=0\}} I(t_j > t_j^o)dt_j}{\int f(t_i^o, \mathbf{t}_{-i}|\Theta)\pi(\Theta)d\Theta \prod_{\{j:\delta_j=0\}} I(t_j > t_j^o)dt_j} \\
&= \frac{f(\mathcal{D}_{-i})}{f(t_i^o, \mathcal{D}_{-i})} = \frac{1}{f(t_i^o|\mathcal{D}_{-i})}
\end{aligned} \tag{C.6}$$

and for $\delta_i = 0$,

$$\begin{aligned}
E_{(\mathbf{t},\Theta|\mathcal{D})}\left\{\frac{1}{S(t_i^o|\mathbf{t}_{-i},\Theta)}\right\} &= \frac{\int \frac{1}{S(t_i^o|\mathbf{t}_{-i},\Theta)} f(\mathbf{t}|\Theta)\pi(\Theta)d\Theta \prod_{\{j:\delta_j=0\}} I(t_j > t_j^o)dt_j}{\int f(\mathbf{t}|\Theta)\pi(\Theta)d\Theta \prod_{\{j:\delta_j=0\}} I(t_j > t_j^o)dt_j} \\
&= \frac{\int \frac{f(t_i|\mathbf{t}_{-i},\Theta)}{S(t_i^o|\mathbf{t}_{-i},\Theta)} f(\mathbf{t}_{-i}|\Theta)\pi(\Theta)d\Theta \prod_{\{j:\delta_j=0\}} I(t_j > t_j^o)dt_j}{\int f(t_i^o|\mathbf{t}_{-i},\Theta) f(\mathbf{t}_{-i}|\Theta)\pi(\Theta)d\Theta \prod_{\{j:\delta_j=0\}} I(t_j > t_j^o)dt_j} \\
&= \frac{\int f(\mathbf{t}_{-i}|\Theta)\pi(\Theta)d\Theta \prod_{\{j\neq i:\delta_j=0\}} I(t_j > t_j^o)dt_j}{\int S(t_i^o|\mathbf{t}_{-i},\Theta) f(\mathbf{t}_{-i}|\Theta)\pi(\Theta)d\Theta \prod_{\{j\neq i:\delta_j=0\}} I(t_j > t_j^o)dt_j} \\
&= \frac{1}{S(t_i^o|\mathcal{D}_{-i})}
\end{aligned} \tag{C.7}$$

Thus it follows that

$$\text{CPO}_i = \left(E \left[\frac{1}{f(t_i^o|\mathbf{t}_{-i}, \Theta)^{\delta_i} S(t_i^o|\mathbf{t}_{-i}, \Theta)^{1-\delta_i}} \right] \right)^{-1}, \tag{C.8}$$

where the expectation E is taken with respect to the joint posterior of $\{\mathbf{t}, \Theta|\mathcal{D}\}$. Here $f(t_i^o|\mathbf{t}_{-i}, \Theta)$ and $S(t_i^o|\mathbf{t}_{-i}, \Theta)$ are given by

$$\begin{aligned} f(t_i^o|\mathbf{t}_{-i}, \Theta) &= \frac{1}{\sigma_{-i}} \phi \left(\frac{\Phi^{-1}\left\{F_{\mathbf{x}_i}(t_i^o|\Theta)\right\} - \mu_{-i}}{\sigma_{-i}} \right) \frac{f_{\mathbf{x}_i}(t_i^o|\Theta)}{\phi\left(\Phi^{-1}\left\{F_{\mathbf{x}_i}(t_i^o|\Theta)\right\}\right)}, \\ S(t_i^o|\mathbf{t}_{-i}, \Theta) &= 1 - \Phi \left\{ \frac{\Phi^{-1}\left\{F_{\mathbf{x}_i}(t_i^o|\Theta)\right\} - \mu_{-i}}{\sigma_{-i}} \right\}, \end{aligned} \tag{C.9}$$

where $\mu_{-i} = -\sum_{j \neq i} C_{ij}^{-} \Phi^{-1}\left\{F_{\mathbf{x}_i}(t_i|\Theta)\right\} / C_{ii}^{-}$ and $\sigma_{-i}^2 = 1/C_{ii}^{-}$ with C_{ij}^{-} being the ijth element of $\mathbf{C}^{-1}$.

4 Web Appendix D: Bayesian Approach to Li and Lin (2006)

4.1 Model Specification

Assume that $T_i|\mathbf{x}_i$ marginally follows the Cox proportional hazard model with cumulative distribution function (cdf)

$$F_{\mathbf{x}_i}(t) = 1 - \exp\left\{-\Lambda_0(t) e^{\mathbf{x}_i'\boldsymbol{\beta}}\right\}$$

and probability density function (pdf)

$$f_{\mathbf{x}_i}(t) = \exp\left\{-\Lambda_0(t) e^{\mathbf{x}_i'\boldsymbol{\beta}}\right\} \lambda_0(t) e^{\mathbf{x}_i'\boldsymbol{\beta}},$$

where $\boldsymbol{\beta}$ is a $p \times 1$ vector of regression coefficients, $\lambda_0(t)$ is the baseline hazard function and $\Lambda_0(t) = \int_0^t \lambda_0(s)ds$ is the cumulative baseline hazard function. The piecewise exponential model provides a flexible framework to deal with the baseline hazard (e.g. Walker and Mallick, 1997; Aslanidou et al., 1998; Qiou et al., 1999). We partition the time period $\mathbb{R}^+$ into M intervals, say $I_k = (d_{k-1}, d_k], k = 1, \ldots, M$, where $d_0 = 0$ and $d_M = \infty$. The baseline hazard is then assumed to be constant within each interval, i.e.

$$\lambda_0(t) = \sum_{k=1}^{M} h_k I\{t \in I_k\},$$

where $h_k \overset{iid}{\sim} \text{Ga}(\nu_0 h, \nu_0)$ are unknown hazard values and $I\{A\}$ is the usual indicator function, i.e. 1 when A is true, 0 otherwise. Consequently, the cumulative baseline hazard function can be written as

$$\Lambda_0(t) = \sum_{k=1}^{M(t)} h_k \Delta_k(t),$$

where $M(t) = \min\{k : d_k \geq t\}$ and $\Delta_k(t) = \min\{d_k, t\} - d_{k-1}$. In fact, the above piecewise exponential model centers $\lambda_0(t)$ at the exponential hazard family $\lambda_h(t) \equiv h$ indexed by h. However the resulting predictive density is not differentiable at the jump points among the M intervals, which is not desirable for many applications and especially for prediction purposes. We propose a mixture of piecewise exponential model by taking the index h to be random; the resulting mixture model yields a differentiable, i.e. smooth, density. Specifically, we set $d_k = F_h^{-1}(k/M), k = 0, \ldots, M$, where $F_h(\cdot)$ is the cdf of exponential distribution with rate parameter h, and put a prior on h, say $h \sim N(h_0, V_0)$. Regardless, after incorporating spatial dependence as described in Section 2 of the main article, we consider the following hierarchical model for the data together with the augmented latent true event-times:

$$
\begin{aligned}
\delta_i | t_i &= I(t_i = t_i^o), \; i = 1, \ldots, n \\
t_i | \boldsymbol{\beta}, \mathbf{h} &\sim F_{\mathbf{x}_i}(t) = 1 - \exp\left\{ -\sum_{k=1}^{M(t)} h_k \Delta_k(t) e^{\mathbf{x}_i' \boldsymbol{\beta}} \right\}, \; i = 1, \ldots, n \\
\mathbf{z} = (z_1, \ldots, z_n)' | \mathbf{t}, \boldsymbol{\beta}, \boldsymbol{\theta} &\sim N_n(\mathbf{0}, \mathbf{C}), \; z_i = \Phi^{-1}\left\{F_{\mathbf{x}_i}(t_i)\right\}, i = 1, \ldots, n \\
h_k &\overset{iid}{\sim} \text{Ga}(r_0 h, r_0), k = 1, \ldots, M, \quad h \sim N(h_0, v_0^2) \\
\boldsymbol{\beta} | \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0 &\overset{iid}{\sim} N_p(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0) \\
(\theta_1, \theta_2) &\sim \text{Beta}(\theta_{1a}, \theta_{1b}) \times \text{Ga}(\theta_{2a}, \theta_{2b})
\end{aligned} \tag{D.10}
$$

where $\mathbf{t} = (t_1, \ldots, t_n)'$, $\mathbf{h} = (h_1, \ldots, h_M)$. We consider following default hyper-prior values: $M = 10$, $r_0 = 1$, $h_0 = \hat{h}$, $\boldsymbol{\mu}_0 = \mathbf{0}$, $\boldsymbol{\Sigma}_0 = 10^5 \mathbf{I}_p$, $\theta_{1a} = \theta_{1b} = \theta_{2a} = \theta_{2b} = 1$, where $\hat{h}$ is the maximum likelihood estimate of the rate parameter h from fitting an exponential PH model. An R function `spCopulaCoxph` calling compiled `C++` to fit this model is provided in the R package `spBayesSurv` accompanying this paper. We also provide a function `indeptCoxph` to fit the non-spatial standard PH model with above baseline specification.

Remark: The function `spCopulaCoxph` provides an option to determine whether the centering parameter h is random or not. For random h, the `spCopulaCoxph` fails to work for certain data sets especially when spatial correlation is large; while the function `indeptCoxph` works without any problem. However, when h is fixed, both functions work very well.

4.2 MCMC

The full likelihood function for the data is given by

$$
\begin{aligned}
\mathcal{L}(\mathbf{t},\mathbf{h},\boldsymbol{\beta},\boldsymbol{\theta}) \propto & \prod_{i=1}^{n} \exp\left\{-\Lambda_0(t_i)e^{\mathbf{x}_i'\boldsymbol{\beta}}\right\} \lambda_0(t_i)e^{\mathbf{x}_i'\boldsymbol{\beta}} \left\{\delta_i I(t_i = t_i^o) + (1-\delta_i)I(t_i > t_i^o)\right\} \\
& \times |\mathbf{C}|^{-1/2}\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}^{-1}-\mathbf{I}_n)\mathbf{z}\right\} \\
& \times \prod_{k=1}^{M} \frac{r_0^{r_0 h}}{\Gamma(r_0 h)} h_k^{r_0 h-1}\exp\{-r_0 h_k\} \times \exp\left\{-\frac{1}{2v_0^2}(h-h_0)^2\right\} \\
& \times \exp\left\{-\frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\mu}_0)'\boldsymbol{\Sigma}_0^{-1}(\boldsymbol{\beta}-\boldsymbol{\mu}_0)\right\} \\
& \times \theta_1^{\theta_{1a}-1}(1-\theta_1)^{\theta_{1b}-1} \times \theta_2^{\theta_{2a}-1}e^{-\theta_{2b}\theta_2}
\end{aligned}
\tag{D.11}
$$

The MCMC sampling steps are as follow:

Step 1: Update h using adaptive Metropolis-Hastings.
The full conditional distribution for h is

$$
\begin{aligned}
f(h|\text{else}) &\propto \prod_{k=1}^{M} \frac{r_0^{r_0 h}}{\Gamma(r_0 h)} h_k^{r_0 h-1} I(h>0)\exp\left\{-\frac{1}{2v_0^2}(h-h_0)^2\right\} \\
&\propto \exp\left\{M r_0 h \log r_0 - M\log\Gamma(r_0 h) + r_0 h\sum_{k=1}^{M}\log(h_k) - \frac{1}{2v_0^2}(h-h_0)^2\right\} I(h>0).
\end{aligned}
$$

Step 2: Update t_i for $i = 1, \ldots, n$.
The full conditional distribution for t_i is

$$
f(t_i|\text{else}) \propto \delta_i I(t_i = t_i^o) + (1-\delta_i) f_{\mathbf{x}_i}(t_i)\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}^{-1}-\mathbf{I}_n)\mathbf{z}\right\} I(t_i > t_i^o).
$$

If $\delta_i = 1$, update $t_i = t_i^o$. If $\delta_i = 0$, propose $t_i^* = F_{\mathbf{x}_i}^{-1}(u_i)$ with u_i from Uniform $(F_{\mathbf{x}_i}(t_i^o), 1)$, and then accept it with probability

$$
\min\left\{1, \frac{\exp\left\{-\frac{1}{2}\mathbf{z}^{*\prime}(\mathbf{C}^{-1}-\mathbf{I}_n)\mathbf{z}^*\right\}}{\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}^{-1}-\mathbf{I}_n)\mathbf{z}\right\}}\right\},
$$

where $\mathbf{z}^* = (z_i^*, \ldots, z_n^*)'$ is the new transformed vector corresponding to t_i^*.

Step 3: Update h_k for $k = 1, \ldots, M$.

The full conditional distribution for h_k is

$$
\begin{aligned}
f(h_k|\text{else}) \propto & \exp\left\{-\sum_{i=1}^{n}\sum_{k=1}^{M(t_i)} h_k \Delta_k(t_i) e^{\mathbf{x}_i'\boldsymbol{\beta}}\right\}\left\{\prod_{\{i:M(t_i)=k\}} h_{M(t_i)}\right\} h_k^{r_0 h-1}\exp\{-r_0 h_k\} \\
& \times \exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}^{-1}-\mathbf{I}_n)\mathbf{z}\right\} \\
\propto & \exp\left\{-h_k \sum_{\{i:M(t_i)\geq k\}} \Delta_k(t_i) e^{\mathbf{x}_i'\boldsymbol{\beta}}\right\}\left\{h_k^{\sum_{i=1}^{n} I\{M(t_i)=k\}}\right\} h_k^{r_0 h-1}\exp\{-r_0 h_k\} \\
& \times \exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}^{-1}-\mathbf{I}_n)\mathbf{z}\right\} \\
\propto & \, h_k^{r_0 h+m_k-1}\exp\left\{-\left(r_0+l_k\right)h_k\right\}\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}^{-1}-\mathbf{I}_n)\mathbf{z}\right\},
\end{aligned}
$$

where

$$
m_k = \sum_{i=1}^{n} I\{M(t_i)=k\} \quad \text{and} \quad l_k = \sum_{\{i:M(t_i)\geq k\}} \Delta_k(t_i) e^{\mathbf{x}_i'\boldsymbol{\beta}}.
$$

Propose h_k^* from $\text{Ga}\left(r_0 h + m_k, r_0 + l_k\right)$ and accept it with probability

$$
\min\left\{1, \frac{\exp\left\{-\frac{1}{2}\mathbf{z}^{*\prime}(\mathbf{C}^{-1}-\mathbf{I}_n)\mathbf{z}^*\right\}}{\exp\left\{-\frac{1}{2}\mathbf{z}'(\mathbf{C}^{-1}-\mathbf{I}_n)\mathbf{z}\right\}}\right\}.
$$

Step 4: Update $\boldsymbol{\beta}$ using adaptive Metropolis-Hastings.

$$
f(\boldsymbol{\beta}|\text{else}) \propto \exp\left\{\sum_{i=1}^{n}\left(-\Lambda_0(t_i)e^{\mathbf{x}_i'\boldsymbol{\beta}} + \mathbf{x}_i'\boldsymbol{\beta}\right) - \frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\mu}_0)'\boldsymbol{\Sigma}_0^{-1}(\boldsymbol{\beta}-\boldsymbol{\mu}_0) - \frac{1}{2}\mathbf{z}'(\mathbf{C}^{-1}-\mathbf{I}_n)\mathbf{z}\right\}
$$

Let $\hat{\boldsymbol{\beta}}$ be the maximum likelihood estimate of $\boldsymbol{\beta}$ from fitting an exponential Cox model and $\hat{S}_0$ be its estimated covariance matrix. Suppose we are currently in time l and have sampled the states $\boldsymbol{\beta}_0, \boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_{l-1}$. We select an index $l_0 > 0$ for the length of an initial period and define

$$
S_l = \begin{cases} \hat{S}_0, & l \leq l_0 \\ \frac{(2.4)^2}{p}(\mathcal{C}_l + 0.05 I_p) & l > l_0. \end{cases}
$$

where $\mathcal{C}_l$ is the sample covariance matrix of $\boldsymbol{\beta}_0, \boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_{l-1}$. Set $\bar{\boldsymbol{\beta}}_l = \frac{1}{l}\sum_{i=0}^{l-1}\boldsymbol{\beta}_i$. Following (Haario et al., 2001), we can use the recursive equations

$$
\bar{\boldsymbol{\beta}}_{l+1} = \frac{l}{l+1}\bar{\boldsymbol{\beta}}_l + \frac{1}{l+1}\boldsymbol{\beta}_l
$$

and

$$
\begin{aligned}
\mathcal{C}_{l+1} = \text{cov}(\boldsymbol{\beta}_0, \boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_l) &= \frac{1}{l}\sum_{i=0}^{l}\boldsymbol{\beta}_i\boldsymbol{\beta}_i' - \frac{l+1}{l}\bar{\boldsymbol{\beta}}_{l+1}\bar{\boldsymbol{\beta}}_{l+1}' \\
&= \frac{l-1}{l}\mathcal{C}_l + \bar{\boldsymbol{\beta}}_l\bar{\boldsymbol{\beta}}_l' - \frac{l+1}{l}\bar{\boldsymbol{\beta}}_{l+1}\bar{\boldsymbol{\beta}}_{l+1}' + \boldsymbol{\beta}_l\boldsymbol{\beta}_l'/l.
\end{aligned}
$$

Table 1: Simulated data – Scenario II. True value, bias of the point estimator (posterior mean), mean (across Monte Carlo replicates) of the posterior standard deviations (MEAN-SD), standard deviation (across Monte Carlo replicates) of the point estimator (SD-MEAN), and Monte Carlo coverage probability for the 95% credible interval (CP) for the spatial correlation parameter $\boldsymbol{\theta}$. The averaged computing time is also presented.

Model	Parameters	True	BIAS	MEAN-SD	SD-MEAN	CP
LDDPM-spatial	θ_1	0.98	-0.020	0.026	0.019	0.98
	θ_2	0.10	0.018	0.027	0.028	0.93
PH-spatial	θ_1	0.98	-0.015	0.023	0.016	0.99
	θ_2	0.10	0.009	0.019	0.023	0.90

It follows that for $l > l_0$

$$S_{l+1} = \frac{l-1}{l} S_l + \frac{(2.4)^2}{2l} \left(l\bar{\boldsymbol{\beta}}_l \bar{\boldsymbol{\beta}}_l' - (l+1)\bar{\boldsymbol{\beta}}_{l+1}\bar{\boldsymbol{\beta}}_{l+1}' + \boldsymbol{\beta}_l \boldsymbol{\beta}_l' + 0.05 I_p \right).$$

We propose $\boldsymbol{\beta}^*$ from $N_p(\boldsymbol{\beta}_{l-1}, S_l)$ and accept it with probability

$$\min\left\{1, \frac{f(\boldsymbol{\beta}^*|\text{else})}{f(\boldsymbol{\beta}_{l-1}|\text{else})}\right\}.$$

Step 5: Update $\boldsymbol{\theta} = (\theta_1, \theta_2)'$ in the same way as Step 9 in Web Appendix A.

5 Web Appendix E: Additional simulations

5.1 Supplements of Simulation – Scenario II

We test the performance of LDDPM-spatial model when the PH assumption is satisfied and compare it with the PH-spatial model. Similarly to Scenario I, we randomly select 400 locations over the spatial region $[0, 40] \times [0, 100]$ and hold out 100 of them for assessing the prediction performance. We then simulate the event times $T(\mathbf{s})$ at these 400 locations from a PH model $F_x(t) = 1 - \exp\{te^{-x}\}$ with the same sample spatial dependence and distribution on x as described in Scenario I. The noninformative censoring times are simulated from a uniform distribution on $(1, 3)$ so that the censoring rate is about $15\% \sim 35\%$.

Table 1 presents the posterior inferences for spatial correlation parameters $\boldsymbol{\theta} = (\theta_1, \theta_2)$, where the bias of corresponding point estimates (i.e. posterior means), the Monte Carlo mean of posterior standard deviation estimates (MEAN-SD), the Monte Carlo standard deviation of point estimates (SD-MEAN), and the Monte Carlo coverage probability of 95% credible intervals (CP) are reported. The results suggest that the point estimates of $\boldsymbol{\theta}$ are almost unbiased under both the LDDPM-spatial and PH-spatial models. The MEAN-SD and SD-MEAN values under the PH-spatial model are fairly close indicating that the posterior standard deviation is an appropriate estimator of the frequentist standard error.

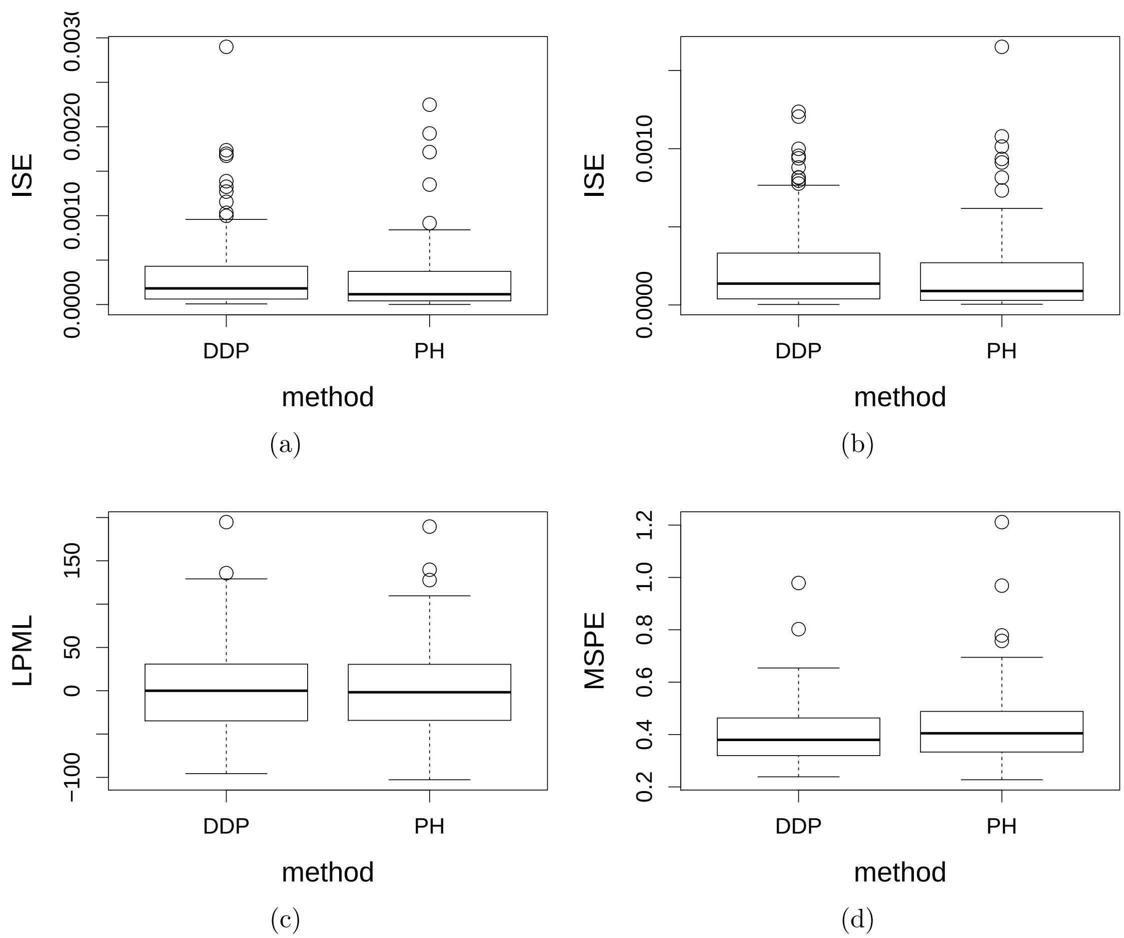

Figure 1: Simulated data – Scenario II. Panel (a) and (b): boxplots of ISEs for fitted survival curves when $x = -1$ and $x = 1$, respectively. Panel (c): boxplots of LPMLs. Panel (d): boxplots of MSPE. In each panel, the two models from left to right are LDDPM-spatial and PH-spatial, respectively.

The CPs are around the nominal 95% level. Overall, even when data are generated from the PH-spatial model, our model still performs reasonably well.

Figure 1 shows boxplots of the ISEs for estimated survival curves, LPMLs, and MSPEs under the considered models. The PH-spatial model provides slightly smaller biases of the fitted survival functions on average, compared with our model. As for prediction ability and accuracy, the results show that two models provide almost the same boxplots of LPMLs and MSPEs, indicating that the LDDP-spatial model is quite competitive even when the PH assumption is satisfied.

6 Web Appendix F: Additional Results to the Analysis of Frog Data

Table 2 shows posterior estimates of the spatial dependence parameters θ_1 and θ_2 under both the LDDPM-spatial and PH-spatial models. Figure 2 presents the Kaplan-Meier survival curves for bdwater=0 versus bdwater=1. The results show that standard parametric or semi-parametric spatial models may be inadequate due to the presence of crossing survivals. Figure 3 presents the trace plots for the correlation parameters $\boldsymbol{\theta}$, which mix reasonably well.

Table 2: Frog data. Posterior statistics for θ_1 and θ_2 under the LDDPM-spatial and PH-spatial models assuming the exponential correlation function. The computing time is also presented.

Model	Parameters	Mean	Median	Std. dev.	95% CI
LDDPM-spatial	θ_1	0.991	0.992	0.004	(0.982, 0.998)
(3.2 hours)	θ_2	0.133	0.130	0.040	(0.060, 0.216)
PH-spatial	θ_1	0.995	0.995	0.002	(0.991, 0.999)
(2.8 hours)	θ_2	0.081	0.080	0.013	(0.059, 0.109)

7 Web Appendix G: Sample R Code for Simulated Data

The R package `spBayesSurv` is available at the website
`http://cran.r-project.org/web/packages/spBayesSurv`.

```
################################################################
# Sample R code for implementing the marginal LDDPM spatial
# survival model proposed by Zhou, Hanson, and Knapp (2015)
# based on a simulated data: mixture of two normals (see
# Section 4 in the paper).
# Provided by Haiming Zhou on 4/2/2015
```

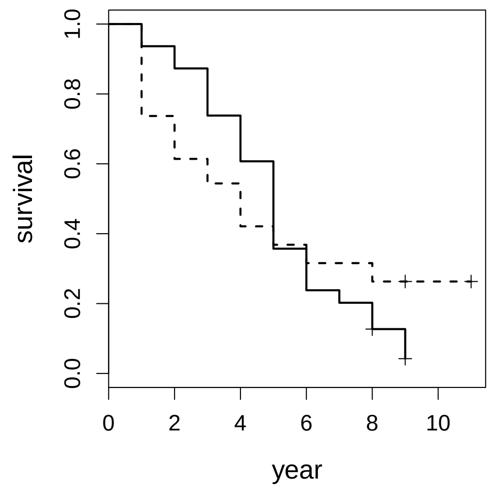

Figure 2: Frog data. Kaplan-Meier survival curves for bdwater=0 (solid) versus bdwater=1 (dashed).

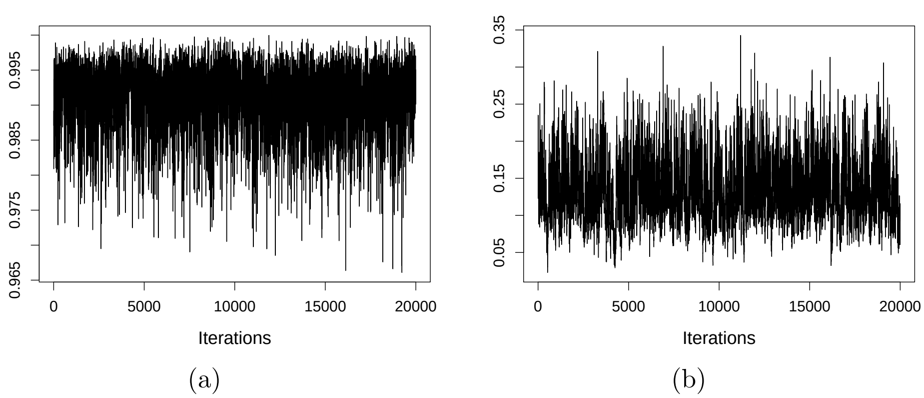

Figure 3: Frog data. Trace plots for θ_1 (panel a) and θ_2 (panel b).

```
###############################################################

##-------------Load libraries-------------------##
rm(list=ls())
library(MASS)
library(Rcpp)
library(RcppArmadillo)
library(coda)
library(survival)
library(spBayesSurv)

##-------------Set the true models--------------##
## True parameters
betaT = cbind(c(3.5, 0.5), c(2.5, -1));
wT = c(0.4, 0.6);
sig2T = c(1^2, 0.5^2);
theta1 = 0.98; theta2 = 0.1;

## The pdf of Ti:
fi = function(y, xi, w=wT){
nw = length(w);
ny = length(y);
res = matrix(0, ny, nw);
Xi = c(1,xi);
for (k in 1:nw){
res[,k] = w[k]*dnorm(y, sum(Xi*betaT[,k]), sqrt(sig2T[k]) )
}
apply(res, 1, sum)
}
## The CDF of Ti:
Fi = function(y, xi, w=wT){
nw = length(w);
ny = length(y);
res = matrix(0, ny, nw);
Xi = c(1,xi);
for (k in 1:nw){
res[,k] = w[k]*pnorm(y, sum(Xi*betaT[,k]), sqrt(sig2T[k]) )
}
apply(res, 1, sum)
}
## The inverse for CDF of Ti
Finvsingle = function(u, xi) {
res = uniroot(function (x) Fi(x, xi)-u, lower=-1e50,
upper=1e50, tol=.Machine$double.eps^0.5);
res$root
}
Finv = function(u, xi) {sapply(u, Finvsingle, xi)};

##-------------Generate data-------------------##
## generate coordinates:
## npred is the # of locations for prediction
n = 300; npred = 30; ntot = n + npred;
ldist = 100; wdist = 40;
s1 = runif(ntot, 0, wdist); s2 = runif(ntot, 0, ldist);
```

```
s = rbind(s1,s2);
#plot(s[1,], s[2,]);
## divide them into blocks
nldist=5; nwdist=2;
nb=nldist*nwdist; nb; # number of blocks;
coor = matrix(0, nb, 4); ## four edges for each block;
tempindex=1; lstep=ldist/nldist; wstep=wdist/nwdist;
for(i in 1:nwdist){
for(j in 1:nldist){
coor[tempindex,] = c((i-1)*wstep, i*wstep, (j-1)*lstep, j*lstep );
tempindex = tempindex + 1;
}
}
## Assign block id for each location
blockid = rep(NA,ntot);
for(i in 1:nb){
blockid[((s1>coor[i,1])*(s1<=coor[i,2])*(s2>coor[i,3])*(s2<=coor[i,4]))==1]=i;
}
## Choose knots S*
nldist=5; nwdist=2;
m=nldist*nwdist; m; # number of knots;
ss = matrix(0, m, 2);
tempindex=1; lstep=ldist/nldist; wstep=wdist/nwdist;
for(i in 1:nwdist){
for(j in 1:nldist){
ss[tempindex,] = c( (i-1)*wstep+wstep/2, (j-1)*lstep+lstep/2);
tempindex = tempindex + 1;
}
}
## Covariance matrix
dnn = .Call("DistMat", s, s, PACKAGE = "spBayesSurv");
corT = theta1*exp(-theta2*dnn)+(1-theta1)*diag(ntot);

## Generate x
x = runif(ntot,-1.5,1.5);
X = cbind(rep(1,ntot), x);
p = ncol(X); # number of covariates + 1
## Generate transformed log of survival times
z = mvrnorm(1, rep(0, ntot), corT);
## Generate log of survival times y
u = pnorm(z);
y = rep(0, ntot);
for (i in 1:ntot){
y[i] = Finv(u[i], x[i]);
}
#plot(x,y);
yTrue = y;

## Censoring scheme
Centime = runif(ntot, 3,4); #Centime = 10000;
delta = (y<=Centime) +0 ;
sum(delta)/ntot; #non-censoring rate
cen = which(delta==0);
y[cen] = Centime[cen];
```

```
## make a data frame
dtotal = data.frame(s1=s1, s2=s2, y=y, x=x, delta=delta,
yTrue=yTrue, id=blockid, t=exp(y));
## Hold out npred=30 for prediction purpose
predindex = sample(1:ntot, npred);
dpred = dtotal[predindex,];
dtrain = dtotal[-predindex,];

# rename the variables
d = dtrain; n=nrow(d); n;
d = d[order(d$id),];
s = cbind(d$s1, d$s2);

# FSA settings
knots = list(ss=ss, blockid=d$id);

# Prediction settings
xpred = dpred$x;
s0 = cbind( dpred$s1, dpred$s2 );
prediction = list(spred=s0, xpred=xpred, predid=dpred$id);

###############################################################
# fit the model using default priors
###############################################################
# MCMC parameters
nburn <- 2000
nsave <- 2000
nskip <- 0
ndisplay <- 500
mcmc <- list(nburn=nburn,
nsave=nsave,
nskip=nskip,
ndisplay=ndisplay)

# Prior information
prior = list(N = 10,
a0 = 2, b0 = 2);

# current state values
state <- NULL;

# Fit the model
ptm <- proc.time()
res = spCopulaDDP( y=d$y, delta=d$delta, x=d$x, s = s, prediction=prediction,
prior=prior, mcmc=mcmc,state=state,FSA=TRUE,knots=knots);
systime = sum((proc.time() - ptm)[1:2]);
# trace plots
par(mfrow = c(3,2))
w.save2 = res$w;
Kindex = which.max(rowMeans(w.save2));
traceplot(mcmc(w.save2[Kindex,]), main="w")
sig2.save2 = res$sigma2;
traceplot(mcmc(sig2.save2[Kindex,]), main="sig2")
```

```
beta.save2 = res$beta;
alpha.save2 = res$alpha;
traceplot(mcmc(beta.save2[2,Kindex,]), main="beta")
traceplot(mcmc(alpha.save2), main="alpha")
theta1.save2 = res$theta1;
theta2.save2 = res$theta2
traceplot(mcmc(theta1.save2), main="theta1")
traceplot(mcmc(theta2.save2), main="theta2")

## LPML
LPML2 = sum(log(res$cpo)); LPML2;
## MSPE
mean((dpred$yTrue-apply(res$Ypred, 1, median))^2);

## Proportions for number of clusters
gg=gg=apply(res$w, 2, function(x) length(which(x>0.001)) );
table(gg)/length(gg);

## plots
par(mfrow = c(2,2));
xnew = c(-1, 1);
xpred = cbind(xnew);
nxpred = nrow(xpred);
ygrid = seq(0,6.0,0.05); tgrid = exp(ygrid);
ngrid = length(ygrid);
estimates = GetCurves(res, xpred, ygrid, CI=c(0.05, 0.95));
fhat = estimates$fhat;
Shat = estimates$Shat;
## density in y
plot(ygrid, fi(ygrid, xnew[1]), "l", lwd=2, ylim=c(0, 0.8),
xlim=c(0,6), main="density in y")
for(i in 1:nxpred){
lines(ygrid, fi(ygrid, xnew[i]), lwd=2)
lines(ygrid, fhat[,i], lty=2, lwd=2, col=4);
}
## survival in y
plot(ygrid, 1-Fi(ygrid, xnew[1]), "l", lwd=2, ylim=c(0, 1),
xlim=c(0,6), main="survival in y")
for(i in 1:nxpred){
lines(ygrid, 1-Fi(ygrid, xnew[i]), lwd=2)
lines(ygrid, Shat[,i], lty=2, lwd=2, col=4);
lines(ygrid, estimates$Shatup[,i], lty=2, lwd=1, col=4);
lines(ygrid, estimates$Shatlow[,i], lty=2, lwd=1, col=4);
}
## density in t
plot(tgrid, fi(ygrid, xnew[1])/tgrid, "l", lwd=2, ylim=c(0, 0.15),
xlim=c(0,100), main="density in t")
for(i in 1:nxpred){
lines(tgrid, fi(ygrid, xnew[i])/tgrid, lwd=2)
lines(tgrid, fhat[,i]/tgrid, lty=2, lwd=2, col=4);
}
## survival in t
plot(tgrid, 1-Fi(ygrid, xnew[1]), "l", lwd=2, ylim=c(0, 1),
xlim=c(0,100), main="survival in t")
```

```
for(i in 1:nxpred){
lines(tgrid, 1-Fi(ygrid, xnew[i]), lwd=2)
lines(tgrid, Shat[,i], lty=2, lwd=2, col=4);
lines(tgrid, estimates$Shatup[,i], lty=2, lwd=1, col=4);
lines(tgrid, estimates$Shatlow[,i], lty=2, lwd=1, col=4);
}
```

8 Web Appendix H: Measures of Dependence

In this section, we explore dependence relations between the original event times in the framework of copula models. Kendall's tau and Spearman's rho are the two most widely used scale-invariant measures for the overall dependence of a pair of subjects over the entire lifespan by integrating over time, both of which are based on a form of dependence known as *concordance* (Nelsen, 2006). In the context of survival data, we say a pair of random event times are concordant if large (small) values of one tend to be associated with large (small) values of the other, otherwise they are discordant. Specifically, when $\mathbf{Y} = (Y_1, \dots, Y_n) = (\log T_1, \dots, \log T_n)$ follows the marginal LDDPM spatial survival model, the Kendall's tau and Spearman's rho of the original event times T_i and T_j (also the same as those of Y_i and Y_j based on their definitions) can be expressed as

$$\tau_{i,j}^K = r\left(\int_0^1 \int_0^1 C_{i,j}(u_i, u_j; \boldsymbol{\theta}) dC_{i,j}(u_i, u_j; \boldsymbol{\theta})\right) - 1 = 4E[C_{i,j}(U_i, U_j; \boldsymbol{\theta})] - 1 \qquad \text{(H.12)}$$

and

$$\rho_{i,j}^S = 12 \int_0^1 \int_0^1 u_i u_j dC_{i,j}(u_i, u_j; \boldsymbol{\theta}) - 3 = 12E[U_i U_j] - 3, \qquad \text{(H.13)}$$

where $C_{i,j}(u_i, u_j; \boldsymbol{\theta})$ is a bivariate marginal of the n-dimensional Gaussian copula function and $(U_i, U_j) \sim C_{i,j}(u_i, u_j; \boldsymbol{\theta})$. Thus the Kendall's tau and Spearman's rho are uniquely determined by the copula function which further depends on the spatial correlation parameters $\boldsymbol{\theta}$. The range of these two measures are between -1 and 1, where the higher the value is, the more concordant the two event times are. Although we don't have closed forms for both τ_{ij}^K and $\rho_{i,j}^S$, we can easily evaluate the expectations in equations (H.12) and (H.13) via posterior simulation (Smith, 2013). Given a set of posterior samples $\{\boldsymbol{\theta}^{(l)}, l = 1, \dots, L\}$, we generate an iterate $(U_i^{(l)}, U_j^{(l)})$ from the bivariate marginal $C_{i,j}(u_i, u_j; \boldsymbol{\theta}^{(l)})$, and then estimate the Kendall's tau and Spearman's rho by

$$\hat{\tau}_{i,j}^K = \frac{4}{L} \sum_{l=1}^{L} C_{i,j}(U_i^{(l)}, U_j^{(l)}; \boldsymbol{\theta}^{(l)}) - 1, \qquad \text{(H.14)}$$

and

$$\hat{\rho}_{i,j}^S = \frac{12}{L} \sum_{l=1}^{L} U_i^{(l)} U_j^{(l)} - 1. \qquad \text{(H.15)}$$

For a given copula, the Kendall's tau and Spearman's rho between a pair of random variables are not necessarily the same. In fact they are often quite different for many families of copulas; see Nelsen (2006) for further illustrations.

References

Aslanidou, H., Dey, D., and Sinha, D. (1998). Bayesian analysis of multivariate survival data using Monte Carlo methods. *Canadian Journal of Statistics*, 26(1):33–48.

Haario, H., Saksman, E., and Tamminen, J. (2001). An adaptive Metropolis algorithm. *Bernoulli*, 7(2):223–242.

Konomi, B. A., Sang, H., and Mallick, B. K. (2014). Adaptive Bayesian nonstationary modeling for large spatial datasets using covariance approximations. *Journal of Computational and Graphical Statistics*, 23:802–929.

Nelsen, R. B. (2006). *An Introduction to Copulas*. Springer, 2nd edition.

Qiou, Z., Ravishanker, N., and Dey, D. (1999). Multivariate survival analysis with positive stable frailties. *Biometrics*, 55(2):637–644.

Sang, H. and Huang, J. Z. (2012). A full scale approximation of covariance functions for large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(1):111–132.

Smith, M. S. (2013). Bayesian approaches to copula modelling. *Bayesian Theory and Applications*, pages 336–358.

Tierney, L. and Mira, A. (1999). Some adaptive Monte Carlo methods for Bayesian inference. *Statistics in Medicine*, 18(1718):2507–2515.

Walker, S. G. and Mallick, B. K. (1997). Hierarchical generalized linear models and frailty models with Bayesian nonparametric mixing. *Journal of the Royal Statistical Society, Series B: Methodological*, 59:845–860.