Applying Max-and-Smooth to the UKCP data

Brynjólfur Gauti Guðrúnar Jónsson

2024-12-18

Introduction

This document describes the implementation of the Max-and-Smooth algorithm for fast approximate Bayesian inference in spatial extreme value analysis of climate projections provided by the UKCP. The algorithm is specifically applied to Generalized Extreme Value (GEV) distributions and is implemented in C++ with R interfaces using Rcpp and RcppEigen as well as Stan.

Package Overview

The `maxandsmooth` R package provides tools for fast approximate Bayesian inference for spatial GEV models. The core of the package is implemented in C++ for efficiency, with R wrappers for ease of use.

Key features of the package include:

- Implementation of the Max-and-Smooth algorithm with Gaussian copula dependence
- Efficient C++ code using automatic differentiation and Eigen
- Spatial modeling of GEV parameters using Stan's efficient HMC sampler
- R interface for easy integration into existing extreme value analysis workflows

Algorithm Description

The Max-and-Smooth algorithm provides a computationally efficient approach to Bayesian inference for spatial extreme value models by decomposing the inference into two steps. This approach is particularly well-suited for spatial GEV models where we have both temporal replicates and spatial dependence.

Overview

Let $Y = \{y_{it}\}$ be observations at locations $i = 1, \ldots, p$ and times $t = 1, \ldots, n$. We model these through GEV marginal distributions with spatially varying parameters and a Gaussian copula dependence structure:

1. **Marginal Distribution**: At each location i:

$$Y_{it} \sim \text{GEV}(\mu_i, \sigma_i, \xi_i)$$

2. **Dependence Structure**: Transform to Gaussian margins via:

$$Z_{it} = \Phi^{-1}(F_{\text{GEV}}(Y_{it}|\mu_i, \sigma_i, \xi_i))$$

where the spatial dependence is captured through a Matérn-like precision structure:

$$Z_t \sim \mathcal{N}(0, Q^{-1}), \quad Q = (Q_{\rho_1} \otimes I_{n_2} + I_{n_1} \otimes Q_{\rho_2})^{\nu+1}$$

Inference Steps

The algorithm proceeds in two main steps:

1. **Max Step**: Joint maximum likelihood estimation

   - Input: Raw observations Y
   - Process:
     a. Transform parameters: $(\psi, \tau, \phi) = (\log \mu, \log \sigma - \log \mu, \text{logit}(\xi))$
     b. Maximize joint log-likelihood combining GEV margins and Gaussian copula
     c. Compute Hessian at MLE for uncertainty quantification
   - Output:
     – MLEs $\hat{\eta} = (\hat{\psi}, \hat{\tau}, \hat{\phi})$
     – Precision matrix $Q_{\eta y}$ (negative Hessian)

2. **Smooth Step**: Spatial smoothing via BYM2 model

   - Input: MLEs $\hat{\eta}$ and precision $Q_{\eta y}$ from Max step
   - Process: For each parameter type $k \in \{\psi, \tau, \phi\}$:
     a. Decompose into spatial and random components:

$$\eta_k = \mu_k \mathbf{1} + \sigma_k(\sqrt{\rho_k/c}\eta_k^{\text{spatial}} + \sqrt{1 - \rho_k}\eta_k^{\text{random}})$$

     b. Apply ICAR prior to spatial component
     c. Sample posterior using MCMC
   - Output: Posterior samples of smoothed parameters

Key Features

The algorithm offers several computational advantages:

1. **Parallel Processing**: The Max step can be parallelized across locations
2. **Dimensionality Reduction**: The Smooth step works with summary statistics (MLEs) rather than raw data
3. **Efficient Sampling**: Uses Stan's NUTS sampler with sparse matrix operations
4. **Uncertainty Propagation**: Incorporates parameter uncertainty through $Q_{\eta y}$

Implementation

The method is implemented using:

1. C++ with automatic differentiation for the Max step:

   - Efficient computation of GEV density and transformations
   - Sparse matrix operations for Gaussian copula likelihood
   - L-BFGS optimization with analytical gradients

2. Stan for the Smooth step:

   - BYM2 spatial model with PC priors
   - Custom functions for sparse precision matrices
   - Efficient HMC sampling

This two-step approach provides a computationally tractable alternative to full MCMC for spatial extreme value models while maintaining proper uncertainty quantification through the entire inference pipeline.

Code Structure

The package is organized into several key files:

1. `src/gev_reverse_mode.cpp`: Implements the Max step (maximum likelihood estimation for GEV) assuming a known Gaussian copula
2. `Stan/stan_smooth_bym2.stan` Implements the Smooth step using Stan

Max Step

The Max step involves computing location-wise maximum likelihood estimates (MLEs) for the GEV model parameters while accounting for spatial dependence through a Matérn-like Gaussian copula structure.

Data Structure and Model Specification

Let Y be an $n \times p$ matrix of observations where:

- Rows $(i = 1, \ldots, n)$ represent temporal replicates
- Columns $(j = 1, \ldots, p)$ represent spatial locations

The model combines GEV marginal distributions with a Gaussian copula:

1. **Marginal GEV distributions**: At each location j, observations follow a GEV distribution:

$$Y_{ij} \sim \text{GEV}(\mu_j, \sigma_j, \xi_j)$$

2. **Spatial dependence**: The dependence structure is captured by transforming the observations to standard normal using the prowbability integral transform:

$$Z_{ij} = \Phi^{-1}(F_{\text{GEV}}(Y_{ij}|\mu_j, \sigma_j, \xi_j))$$

where F_{GEV} is the GEV CDF and Φ^{-1} is the standard normal quantile function.

3. **Matérn-like precision structure**: The transformed observations follow a multivariate normal distribution with precision matrix:

$$Q = (Q_{\rho_1} \otimes I_{n_2} + I_{n_1} \otimes Q_{\rho_2})^{\nu+1}$$

where:

- Q_ρ is the precision matrix of a standardized AR(1) process
- $\otimes$ denotes the Kronecker product
- ν is a smoothness parameter
- The matrix is scaled to ensure unit marginal variances

Log-likelihood Function

The total log-likelihood combines the GEV marginal contributions and the Gaussian copula:

$$\ell(\theta|Y) = \sum_{j=1}^{p} \sum_{i=1}^{n} \ell_{\text{GEV}}(y_{ij}|\mu_j, \sigma_j, \xi_j) + \ell_{\text{copula}}(Z|Q)$$

where:

1. The GEV log-likelihood for a single observation is:

$$\ell_{\text{GEV}}(y|\mu, \sigma, \xi) = -\log \sigma - (1 + \frac{1}{\xi})\log(1 + \xi\frac{y-\mu}{\sigma}) - (1 + \xi\frac{y-\mu}{\sigma})^{-1/\xi}$$

2. The Gaussian copula log-likelihood is:

$$\ell_{\text{copula}}(Z|Q) = \frac{1}{2}\log|Q| - \frac{1}{2}Z^T Q Z + \frac{1}{2}Z^T Z$$

where the last term accounts for the standard normal margins.

Implementation Details

The optimization is performed using automatic differentiation and the L-BFGS algorithm. Key implementation features include:

1. **Parameter transformations**:

$$(\psi, \tau, \phi) = (\log\mu, \log\sigma - \log\mu, \text{logit}(\xi))$$

2. **Efficient computation** of the quadratic form $Z^T Q Z$ and log-determinant $\log|Q|$ by exploiting the Kronecker structure of the precision matrix

3. **Probability integral transform** using accurate approximations to the GEV CDF and normal quantile function

4. **Automatic differentiation** (using autodiff's reverse mode) for accurate gradient and Hessian computation

Smooth Step

The Smooth step performs Bayesian inference on the latent parameter fields using the maximum likelihood estimates from the Max step as noisy observations. We implement this using Stan's efficient Hamiltonian Monte Carlo sampler with a BYM2 (Besag-York-Mollié) spatial model.

Model Structure

Let $\hat{\eta}$ be the vector of maximum likelihood estimates from the Max step, arranged as:

$$\hat{\eta} = (\hat{\psi}_1, \dots, \hat{\psi}_p, \hat{\tau}_1, \dots, \hat{\tau}_p, \hat{\phi}_1, \dots, \hat{\phi}_p)^T$$

where p is the number of spatial locations and $(\hat{\psi}, \hat{\tau}, \hat{\phi})$ represent the transformed GEV parameters.

Spatial Random Effects

For each parameter type $k \in \{\psi, \tau, \phi\}$, we decompose the spatial variation into structured and unstructured components following the BYM2 parameterization:

$$\eta_k = \mu_k \mathbf{1} + \sigma_k \left(\sqrt{\frac{\rho_k}{c}} \eta_k^{\text{spatial}} + \sqrt{1 - \rho_k} \eta_k^{\text{random}} \right)$$

where:

- μ_k is the overall mean
- $\sigma_k > 0$ is the marginal standard deviation
- $\rho_k \in [0, 1]$ is the mixing parameter controlling the balance between spatial and unstructured variation
- c is a scaling factor that ensures the marginal variance of the spatial component is approximately 1
- η_k^{spatial} follows an intrinsic conditional autoregressive (ICAR) prior
- $\eta_k^{\text{random}} \sim \mathcal{N}(0, I)$ represents unstructured random effects

ICAR Prior Specification

The ICAR prior for the spatial component η_k^{spatial} is specified through its full conditional distributions:

$$\eta_{k,i}^{\text{spatial}} | \eta_{k,-i}^{\text{spatial}} \sim \mathcal{N} \left(\frac{1}{n_i} \sum_{j \sim i} \eta_{k,j}^{\text{spatial}}, \frac{1}{n_i} \right)$$

where $j \sim i$ indicates that locations i and j are neighbors, and n_i is the number of neighbors for location i. This is implemented efficiently in Stan through the sum of squared differences form:

$$p(\eta_k^{\text{spatial}}) \propto \exp \left(-\frac{1}{2} \sum_{i \sim j} (\eta_{k,i}^{\text{spatial}} - \eta_{k,j}^{\text{spatial}})^2 \right)$$

with an additional soft sum-to-zero constraint implemented via $\sum_i \eta_{k,i}^{\text{spatial}} \sim \mathcal{N}(0, 0.001p)$.

Observation Model

The observation model links the MLEs to the latent field through a multivariate normal distribution:

$$\hat{\eta}|\eta \sim \mathcal{N}(\eta, Q_{\eta y}^{-1})$$

where $Q_{\eta y}$ is the precision matrix obtained from the negative Hessian in the Max step. To handle this efficiently in Stan, we:

1. Pre-compute the Cholesky factor L of $Q_{\eta y} = LL^T$
2. Store L in a sparse format using arrays of indices and values
3. Implement a custom log probability function that computes:

$$\log p(\hat{\eta}|\eta) = \frac{1}{2}\log|Q_{\eta y}| - \frac{1}{2}(\hat{\eta} - \eta)^T Q_{\eta y}(\hat{\eta} - \eta)$$

   using the sparse Cholesky representation

Prior Distributions

We specify weakly informative priors:

$$\sigma_k \sim \text{Exponential}(1)$$
$$\rho_k \sim \text{Beta}(1,1)$$
$$\mu_k \sim \text{flat}$$

for each parameter type k. The exponential prior on σ_k provides weak regularization while ensuring positivity, while the uniform Beta prior on ρ_k allows the data to determine the balance between spatial and unstructured variation.

Posterior Inference

The model is fit using Stan's implementation of the No-U-Turn Sampler (NUTS), providing:

1. Posterior samples of the latent field η
2. Uncertainty quantification through the posterior distributions of μ_k, σ_k, and ρ_k
3. Decomposition of spatial variation through the posterior distributions of η_k^{spatial} and η_k^{random}