University of Innsbruck

BAMLSS: Bayesian additive models for location, scale and shape (and beyond)

Nikolaus Umlauf, Nadja Klein, Achim Zeileis

Working Papers in Economics and Statistics

2017-04

Research Platform
eeecon
Empirical and Experimental Economics
University of Innsbruck
http://eeecon.uibk.ac.at/

University of Innsbruck
Working Papers in Economics and Statistics

The series is jointly edited and published by

- Department of Banking and Finance
- Department of Economics
- Department of Public Finance
- Department of Statistics

Contact address of the editor:
Research platform “Empirical and Experimental Economics”
University of Innsbruck
Universitaetsstrasse 15
A-6020 Innsbruck
Austria
Tel: + 43 512 507 7171
Fax: + 43 512 507 2970
E-mail: eeecon@uibk.ac.at

The most recent version of all working papers can be downloaded at
http://eeecon.uibk.ac.at/wopec/

For a list of recent papers see the backpages of this paper.

BAMLSS: Bayesian Additive Models for Location, Scale and Shape (and Beyond)

Nikolaus Umlauf
Universität Innsbruck

Nadja Klein
Universität Göttingen

Achim Zeileis
Universität Innsbruck

Abstract

Bayesian analysis provides a convenient setting for the estimation of complex generalized additive regression models (GAMs). Since computational power has tremendously increased in the past decade it is now possible to tackle complicated inferential problems, e.g., with Markov chain Monte Carlo simulation, on virtually any modern computer. This is one of the reasons why Bayesian methods have become increasingly popular, leading to a number of highly specialized and optimized estimation engines and with attention shifting from conditional mean models to probabilistic distributional models capturing location, scale, shape (and other aspects) of the response distribution. In order to embed many different approaches suggested in literature and software, a unified modeling architecture for distributional GAMs is established that exploits the general structure of these models and encompasses many different response distributions, estimation techniques (posterior mode or posterior mean), and model terms (fixed, random, smooth, spatial, ...). It is shown that within this framework implementing algorithms for complex regression problems, as well as the integration of already existing software, is relatively straightforward. The usefulness is emphasized with two complex and computationally demanding application case studies: a large daily precipitation climatology based on more than 1.2 million observations from more than 50 meteorological stations, as well as a Cox model for continuous time with space-time interactions on a data set with over five thousand "individuals".

Keywords: GAMLSS, distributional regression, MCMC, BUGS, R, software.

1. Introduction

The generalized additive model for location, scale and shape (GAMLSS, Rigby and Stasinopoulos 2005) relaxes the distributional assumptions of a response variable in a way that allows for modeling the mean (location) as well as higher moments (scale and shape) in terms of covariates. This is especially useful in cases where, e.g., the response does not follow the exponential family or particular interest lies on scale and shape parameters. Moreover, covariate effects can have flexible forms such as, e.g., linear, nonlinear, spatial or random effects. Hence, each parameter of the distribution is linked to an additive predictor in similar fashion as for the well established generalized additive model (GAM, Hastie and Tibshirani 1990).

The terms of an additive predictor are most commonly represented by basis function approaches. This leads to a very generic model structure and can be further exploited because each term can be transformed into a mixed model representation (Ruppert, Wand, and Carrol 2003; Wand 2003). In a fully Bayesian setting this generality remains because priors on

parameters can also be formalized in a general way, e.g., by assigning normal priors to the regression coefficients of smooth terms (Brezger and Lang 2006; Fahrmeir, Kneib, Lang, and Marx 2013).

The fully Bayesian approach using Markov chain Monte Carlo (MCMC) simulation techniques is particularly attractive since the inferential framework provides valid credible intervals for estimators in situations where confidence intervals for corresponding maximum likelihood estimators based on asymptotic properties fail. This is specifically the case in more complex models, e.g., with response distributions outside the exponential family or when multiple predictors contain several smooth effects (Klein, Kneib, and Lang 2015b). In addition, extensions such as variable selection, non-standard priors for hyper-parameters, or multilevel models are easily included. Due to this and due to the tremendous increase in computational power over the past decade, the number of both, Bayesian and frequentist, estimation engines for such complicated inferential problems has been receiving increasing attention. Existing estimation engines already provide infrastructures for a number of regression problems exceeding univariate responses, e.g., for multinomial, multivariate normal, censored, or truncated response variables, etc. In addition, most of the engines support random effect estimation which can in principle also be utilized for setting up complex models with additive predictors (see, e.g., Wood 2006, Wood 2016a).

However, the majority of engines use different model setups and output formats, which makes it difficult for practitioners, e.g., to compare properties of different algorithms or to select the appropriate distribution and variables, etc. The reasons are manifold: the use of different model specification languages like BUGS (Lunn, Spiegelhalter, Thomas, and Best 2009) or R (R Core Team 2016); different standalone statistical software packages like **BayesX** (Belitz, Brezger, Kneib, Lang, and Umlauf 2017; Umlauf, Adler, Kneib, Lang, and Zeileis 2015), **JAGS** (Plummer 2003), **Stan** (Carpenter *et al.* 2017) or **WinBUGS** (Lunn, Thomas, Best, and Spiegelhalter 2000); or even differences within the same environment, e.g., the R packages **mgcv** (Wood 2016b), **gamlss** (Stasinopoulos and Rigby 2016) and **VGAM** (Yee 2015) implement all model term infrastructures in their own fashion style. This is particularly problematic if all packages are loaded into R's global environment, because some functions that are supposed to fulfill the same purpose have different interpretations.

In this article we present a unified conceptional "Lego toolbox" for complex regression models. We show that iterative estimation algorithms, e.g., for posterior mode or mean estimation based on MCMC simulation, exhibit very similar structures such that the process of model building becomes relatively straightforward, since the model architecture is only a combination of single "bricks". Due to many parallels to the GAMLSS class, the conceptional framework is called BAMLSS (*Bayesian additive models for location, scale and shape*). However, it also encompasses many more general model terms *beyond* linear combinations in a design matrix with regression coefficients. The toolbox can be exploited in three ways: First, to quickly develop new models and algorithms, secondly, to compare existing algorithms and samplers, and third to easily integrate existing implementations. A proof of concept is given in the corresponding R implementation **bamlss** (Umlauf, Klein, Zeileis, and Köhler 2017).

The remainder of the paper is structured as follows. In Section 2 the models supported by this framework are briefly introduced. Section 3 presents the conceptional algorithm design used to estimate numerous (possibly) complex models. In Section 4 the details of the "Lego bricks" that form the estimation systems are presented. Then, Section 5 describes computational strategies for the implementation of the framework. Finally, Section 6 illustrates the concept

using two complex and computationally demanding illustrations: a large climatology model for daily precipitation observations using censored heteroscedastic regression and a Cox model for continuous time with space-time interactions.

2. Model structure

Based on data for $i = 1, \ldots, n$ observations, the models discussed in this paper assume conditional independence of individual response observations given covariates. As in the classes of GAMLSS (Rigby and Stasinopoulos 2005) or distributional regression models (Klein, Kneib, Lang, and Sohn 2015c) all parameters of the response distribution can be modeled by explanatory variables such that

$$y \sim \mathcal{D}\left(h_1(\theta_1) = \eta_1,\ h_2(\theta_2) = \eta_2, \ldots,\ h_K(\theta_K) = \eta_K\right),$$

where $\mathcal{D}$ denotes a parametric distribution for the response variable y with K parameters θ_k, $k = 1, \ldots, K$, that are linked to additive predictors using known monotonic and twice differentiable functions $h_k(\cdot)$. Note that the response may also be a q-dimensional vector $\mathbf{y} = (y_1, \ldots, y_q)^\top$, e.g., when $\mathcal{D}$ is a multivariate distribution (see, e.g., Klein, Kneib, Klasen, and Lang 2015a). The k-th additive predictor is given by

$$\eta_k = \eta_k(\mathbf{x}; \boldsymbol{\beta}_k) = f_{1k}(\mathbf{x}; \boldsymbol{\beta}_{1k}) + \ldots + f_{J_k k}(\mathbf{x}; \boldsymbol{\beta}_{J_k k}), \tag{1}$$

with unspecified (possibly nonlinear) functions $f_{jk}(\cdot)$ of subvectors of a vector $\mathbf{x}$ collecting all available covariate information, $j = 1, \ldots, J_k$ and $k = 1, \ldots, K$ and $\boldsymbol{\beta}_k = (\boldsymbol{\beta}_{1k}, \ldots, \boldsymbol{\beta}_{J_k k})^\top$ are parameters, typically regression coefficients that need to estimated from the data. The vector of function evaluations $\mathbf{f}_{jk} = (f_{jk}(\mathbf{x}_1; \boldsymbol{\beta}_{jk}), \ldots, f_{jk}(\mathbf{x}_n; \boldsymbol{\beta}_{jk}))^\top$ of the $i = 1, \ldots, n$ observations is then given by

$$\mathbf{f}_{jk} = \begin{pmatrix} f_{jk}(\mathbf{x}_1; \boldsymbol{\beta}_{jk}) \\ \vdots \\ f_{jk}(\mathbf{x}_n; \boldsymbol{\beta}_{jk}) \end{pmatrix} = f_{jk}(\mathbf{X}_{jk}; \boldsymbol{\beta}_{jk}), \tag{2}$$

where $\mathbf{X}_{jk}$ $(n \times m_{jk})$ is a design matrix and the structure of $\mathbf{X}_{jk}$ only depends on the type of covariate(s) and prior assumptions about $f_{jk}(\cdot)$. In this notation the k-th parameter vector is given by

$$h_k(\boldsymbol{\theta}_k) = \boldsymbol{\eta}_k = \eta_k(\mathbf{X}_k; \boldsymbol{\beta}_k) = \mathbf{f}_{1k} + \ldots + \mathbf{f}_{J_k k},$$

where $\mathbf{X}_k = (\mathbf{X}_{1k}, \ldots, \mathbf{X}_{J_k k})^\top$ is the combined design matrix for the k-th parameter.

While functions $f_{jk}(\cdot)$ are usually based on a basis function approach, where η_k then is a typical GAM-type or so-called structured additive predictor (STAR, Fahrmeir, Kneib, and Lang 2004; Brezger and Lang 2006), in this paper we relax this assumption and let $f_{jk}(\cdot)$ be an unspecified composition of covariate data $\mathbf{x}$ and regression coefficients $\boldsymbol{\beta}_{jk}$ $(q_{jk} \times 1)$. In the case where it is derived through a basis function approach, it can be written as

$$\mathbf{f}_{jk} = \mathbf{X}_{jk}\boldsymbol{\beta}_{jk},$$

But more general and complex terms are also allowed in the BAMLSS framework. A simple example for a $f_{jk}(\cdot)$ that is nonlinear in the parameters $\boldsymbol{\beta}_{jk}$ would be a Gompertz growth curve

$$\mathbf{f}_{jk} = \beta_1 \cdot \exp\left(-\exp\left(\beta_2 + \mathbf{X}_{jk}\beta_3\right)\right).$$

Note that using basis functions the individual model components $\mathbf{X}_{jk}\boldsymbol{\beta}_{jk}$ can be further decomposed into a mixed model representation given by

$$\mathbf{f}_{jk} = \tilde{\mathbf{X}}_{jk}\tilde{\boldsymbol{\gamma}}_{jk} + \mathbf{U}_{jk}\tilde{\boldsymbol{\beta}}_{jk}, \tag{3}$$

where $\tilde{\boldsymbol{\gamma}}_{jk}$ represents the fixed effects parameters and $\tilde{\boldsymbol{\beta}}_{jk} \sim \mathcal{N}(\mathbf{0}, \tau^2_{jk}\mathbf{I})$ i.i.d. random effects. The design matrix $\mathbf{U}_{jk}$ is derived from a spectral decomposition of the penalty matrix $\mathbf{K}_{jk}$ and $\tilde{\mathbf{X}}_{jk}$ by finding a basis of the null space of $\mathbf{K}_{jk}$ such that $\tilde{\mathbf{X}}_{jk}^\top \mathbf{K}_{jk} = \mathbf{0}$, i.e., parameters $\tilde{\boldsymbol{\gamma}}_{jk}$ are not penalized (see, e.g., Ruppert *et al.* 2003; Wand 2003; Wood 2004; Fahrmeir *et al.* 2013). Such transformations can be used to estimate functions $f_{jk}(\cdot)$ using standard algorithms for random effects (see, e.g., Wood 2016a).

3. A conceptional Lego toolbox

3.1. Response and posterior distribution

The main building block of regression model algorithms is the probability density function $d_y(\mathbf{y}|\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K)$, or for computational reasons its logarithm. Estimation typically requires to evaluate the log-likelihood function

$$\ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X}) = \sum_{i=1}^{n} \log\, d_y(y_i; \theta_{i1} = h_1^{-1}(\eta_{i1}(\mathbf{x}_i; \boldsymbol{\beta}_1)), \ldots, \theta_{iK} = h_K^{-1}(\eta_{iK}(\mathbf{x}_i; \boldsymbol{\beta}_K)))$$

a number of times, where the vector $\boldsymbol{\beta} = (\boldsymbol{\beta}_1^\top, \ldots, \boldsymbol{\beta}_K^\top)^\top$ comprises all regression coefficients/parameters that should be estimated, $\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_K)$ are the respective covariate matrices whose i-th row is denoted $\mathbf{x}_i$ and $\boldsymbol{\theta}_k$ are distribution parameter vectors of length n. Assigning prior distributions $p_{jk}(\cdot)$ to the individual model components results in the log-posterior

$$\log\, \pi(\boldsymbol{\beta}, \boldsymbol{\tau}; \mathbf{y}, \mathbf{X}, \boldsymbol{\alpha}) \propto \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X}) + \sum_{k=1}^{K} \sum_{j=1}^{J_k} \left[\log\, p_{jk}(\boldsymbol{\beta}_{jk}; \boldsymbol{\tau}_{jk}, \boldsymbol{\alpha}_{jk})\right], \tag{4}$$

where $\boldsymbol{\tau} = (\boldsymbol{\tau}_1^\top, \ldots, \boldsymbol{\tau}_K^\top)^\top = (\boldsymbol{\tau}_{11}^\top, \ldots, \boldsymbol{\tau}_{J_1 1}^\top, \ldots, \boldsymbol{\tau}_{1K}^\top, \ldots, \boldsymbol{\tau}_{J_K K}^\top)^\top$ is the vector of all assigned hyper-parameters used within prior functions $p_{jk}(\cdot)$ and similarly $\boldsymbol{\alpha}$ is the set of all fixed prior specifications. More precisely, the rather general prior for the jk-th model term is given by

$$p_{jk}(\boldsymbol{\beta}_{jk}; \boldsymbol{\tau}_{jk}, \boldsymbol{\alpha}_{jk}) \propto d_{\boldsymbol{\beta}_{jk}}(\boldsymbol{\beta}_{jk} | \, \boldsymbol{\tau}_{jk}; \boldsymbol{\alpha}_{\boldsymbol{\beta}_{jk}}) \cdot d_{\boldsymbol{\tau}_{jk}}(\boldsymbol{\tau}_{jk} | \, \boldsymbol{\alpha}_{\boldsymbol{\tau}_{jk}}), \tag{5}$$

with prior densities (or combinations of densities) $d_{\boldsymbol{\beta}_{jk}}(\cdot)$ and $d_{\boldsymbol{\tau}_{jk}}(\cdot)$ that depend on the type of covariate and prior assumptions about $f_{jk}(\cdot)$. In this framework, $\boldsymbol{\tau}_{jk}$ are typically variances, e.g., that account for the degree of smoothness of $f_{jk}(\cdot)$ or the amount of correlation between observations. For example, using a spline representation of $f_{jk}(\cdot)$ in combination with a normal prior for $d_{\boldsymbol{\beta}_{jk}}(\cdot)$ the variances can be interpreted as the inverse smoothing parameters in a penalized regression context, i.e., from a frequentist perspective (4) can be viewed as a penalized log-likelihood. In addition, the fixed prior specifications $\boldsymbol{\alpha}_{jk} = \{\boldsymbol{\alpha}_{\boldsymbol{\beta}_{jk}}, \boldsymbol{\alpha}_{\boldsymbol{\tau}_{jk}}\}$ can further control the shape of $d_{\boldsymbol{\beta}_{jk}}(\cdot)$ and $d_{\boldsymbol{\tau}_{jk}}(\cdot)$, incorporate prior beliefs about $\boldsymbol{\beta}_{jk}$, or for GAM-type models $\boldsymbol{\alpha}_{jk}$ usually holds the so-called penalty matrices, amongst others.

3.2. Model fitting

Bayesian point estimates of $\boldsymbol{\beta}$ and $\boldsymbol{\tau}$ are typically obtained by either one of:

E1. Maximization of the log-posterior for posterior mode estimation.

E2. Solving high-dimensional integrals, e.g., for posterior mean or median estimation.

For the possibly very complex models within the BAMLSS framework, problems E1 and E2 are commonly solved by computer intensive iterative algorithms, since analytical solutions are available only in a few special cases. In either case, the algorithms perform an updating scheme of type

$$(\boldsymbol{\beta}^{(t+1)}, \boldsymbol{\tau}^{(t+1)}) = U(\boldsymbol{\beta}^{(t)}, \boldsymbol{\tau}^{(t)}; \mathbf{y}, \mathbf{X}, \boldsymbol{\alpha}), \tag{6}$$

where function $U(\cdot)$ is an updating function, e.g., for generating one Newton-Raphson step in E1 or getting the next step in an MCMC simulation in E2, amongst others. The updating scheme can be partitioned into separate updating equations using leapfrog or zigzag iteration (see, e.g., Smyth 1996). Now let

$$\begin{aligned}
(\boldsymbol{\beta}_1^{(t+1)}, \boldsymbol{\tau}_1^{(t+1)}) &= U_1(\boldsymbol{\beta}_1^{(t)}, \boldsymbol{\beta}_2^{(t)}, \ldots, \boldsymbol{\beta}_K^{(t)}, \boldsymbol{\tau}_1^{(t)}, \boldsymbol{\tau}_2^{(t)}, \ldots, \boldsymbol{\tau}_K^{(t)}; \mathbf{y}, \mathbf{X}_1, \boldsymbol{\alpha}_1) \\
(\boldsymbol{\beta}_2^{(t+1)}, \boldsymbol{\tau}_2^{(t+1)}) &= U_2(\boldsymbol{\beta}_1^{(t+1)}, \boldsymbol{\beta}_2^{(t)}, \ldots, \boldsymbol{\beta}_K^{(t)}, \boldsymbol{\tau}_1^{(t+1)}, \boldsymbol{\tau}_2^{(t)}, \ldots, \boldsymbol{\tau}_K^{(t)}; \mathbf{y}, \mathbf{X}_2, \boldsymbol{\alpha}_2) \\
&\vdots \\
(\boldsymbol{\beta}_K^{(t+1)}, \boldsymbol{\tau}_K^{(t+1)}) &= U_K(\boldsymbol{\beta}_1^{(t+1)}, \boldsymbol{\beta}_2^{(t+1)}, \ldots, \boldsymbol{\beta}_K^{(t)}, \boldsymbol{\tau}_1^{(t+1)}, \boldsymbol{\tau}_2^{(t+1)}, \ldots, \boldsymbol{\tau}_K^{(t)}; \mathbf{y}, \mathbf{X}_K, \boldsymbol{\alpha}_K)
\end{aligned} \tag{7}$$

be a partitioned updating scheme with updating functions $U_k(\cdot)$, i.e., in each iteration updates for the k-th parameter are computed while holding all other parameters fixed. Furthermore, this strategy can be applied for all terms within a parameter

$$(\boldsymbol{\beta}_{jk}^{(t+1)}, \boldsymbol{\tau}_{jk}^{(t+1)}) = U_{jk}(\boldsymbol{\beta}_{jk}^{(t)}, \boldsymbol{\tau}_{jk}^{(t)}, \cdot) \qquad j = 1, \ldots, J_k, \quad k = 1, \ldots, K, \tag{8}$$

and $U_{jk}(\cdot)$ is an updating function for a single model term.

The partitioned updating system allows for having different functions $U_{jk}(\cdot)$ for different model terms, e.g., in problem E1 some updating functions could be based on iteratively weighted least squares (IWLS, Gamerman 1997) and some on ordinary Newton-Raphson steps (see, e.g., example Section 6.2). In problem E2 using MCMC simulation it is common to mix between several sampling methods depending on the type of model term or distribution parameter.

Using highly modular systems like (7) and (8) it is possible to develop a generic estimation algorithm for numerous possibly very complex models, which is outlined in Algorithm A1. The algorithm starts by initializing all model parameters and predictors. Then an outer iteration loops over all distributional parameters performing an inner iteration updating all model terms of the respective parameter, i.e., the algorithm uses backfitting type updating schemes. In practice, for full Bayesian inference the algorithm is applied twice, i.e., first computing estimates for E1 and then using these as starting values for solving E2.

Finding good starting values is especially important for complex model terms, e.g., for multi-dimensional functions $f_{jk}(\cdot)$ that have multiple smoothing variances in prior densities $p_{jk}(\cdot)$. Therefore, we propose to estimate parameters $\boldsymbol{\tau}_{jk}$ using a goodness-of-fit criterion within the stepwise selection approach presented in Algorithm A2a, similar to Belitz and Lang (2008).

Algorithm A1 Generic BAMLSS model fitting algorithm.

Input: $\mathbf{y}$, $\mathbf{X}$, $\boldsymbol{\alpha}$.
Set: Stopping criterion ε, number of iterations T, e.g., $\varepsilon = 0.0001$, $T = 1000$.
Initialize: $\boldsymbol{\beta}$, $\boldsymbol{\eta}$, $\boldsymbol{\tau}$, e.g., $\boldsymbol{\beta} = \mathbf{0}, \boldsymbol{\tau} = 0.001 \cdot \mathbf{1}$, $\Delta = \varepsilon + 1$, $t = 1$.
  while $(\Delta > \varepsilon)\,\&\,(t < T)$ **do**
    Set $\mathring{\boldsymbol{\eta}} = \boldsymbol{\eta}^{(t)}$.
    for $k = 1$ to K **do**
      for $j = 1$ to J_k **do**
        Obtain new state $(\boldsymbol{\beta}_{jk}^{(t+1)}, \boldsymbol{\tau}_{jk}^{(t+1)}) \leftarrow U_{jk}(\boldsymbol{\beta}_{jk}^{(t)}, \boldsymbol{\tau}_{jk}^{(t)}, \cdot)$ using Algorithm A2a or A2b.
        Compute $\mathbf{f}_{jk}^{(t+1)} \leftarrow f_{jk}(\mathbf{X}_{jk}, \boldsymbol{\beta}_{jk}^{(t+1)})$.
        Update $\boldsymbol{\eta}_k^{(t+1)} \leftarrow \boldsymbol{\eta}_k^{(t)} - \mathbf{f}_{jk}^{(t)} + \mathbf{f}_{jk}^{(t+1)}$.
      end for
    end for
    Compute $\Delta \leftarrow \text{rel.change}(\mathring{\boldsymbol{\eta}}, \boldsymbol{\eta}^{(t+1)})$.
    Increase $t \leftarrow t + 1$.
  end while
Output: Posterior mode estimates $\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}^{(t)}$, $\hat{\boldsymbol{\tau}} = \boldsymbol{\tau}^{(t)}$ for E1;
  or MCMC samples $\boldsymbol{\beta}^{(t)}$, $\boldsymbol{\tau}^{(t)}$, $t = 1, \ldots, T$ for E2.

Algorithm A2a Posterior mode updating $U_{jk}(\cdot)$ with smoothing variance selection.

Input: $\mathbf{y}$, $\mathbf{X}$, $\boldsymbol{\alpha}$, $\boldsymbol{\beta}^{(t)}$, $\boldsymbol{\tau}^{(t)}$.
Set: Goodness-of-fit criterion C.
  for $l = 1$ to L_{jk} **do**
    Set search interval for $\tau_{ljk}^{(t+1)}$, e.g., $\mathcal{I}_{ljk} = [\tau_{ljk}^{(t)} \cdot 10^{-1}, \tau_{ljk}^{(t)} \cdot 10]$.
    Find $\tau_{ljk}^{(t+1)} \leftarrow \underset{\tau_{ljk}^{\star} \in \mathcal{I}_{ljk}}{\arg\min}\, C(U_{jk}(\boldsymbol{\beta}_{jk}^{(t)}, \tau_{ljk}^{\star}, \cdot))$.
  end for
  Update $\boldsymbol{\beta}_{jk}^{(t+1)} \leftarrow U_{jk}(\boldsymbol{\beta}_{jk}^{(t)}, \boldsymbol{\tau}_{jk}^{(t+1)}, \cdot)$.
Output: Updates $\boldsymbol{\beta}_{jk}^{(t+1)}$, $\boldsymbol{\tau}_{jk}^{(t+1)}$.

Algorithm A2b MCMC updating $U_{jk}(\cdot)$.

Input: $\mathbf{y}$, $\mathbf{X}$, $\boldsymbol{\alpha}$, $\boldsymbol{\beta}^{(t)}$, $\boldsymbol{\tau}^{(t)}$.
Set: Sampling method, e.g., derivative-based MCMC (see Section 4.2).
  Sample $\boldsymbol{\beta}_{jk}^{\star} \leftarrow q_{jk}(\boldsymbol{\beta}_{jk}^{\star} \,|\, \boldsymbol{\beta}_{jk}^{(t)})$.
  Compute acceptance probability $\alpha\left(\boldsymbol{\beta}_{jk}^{\star} \,|\, \boldsymbol{\beta}_{jk}^{(t)}\right)$.
  if uniform draw $U(0,1) \leq \alpha\left(\boldsymbol{\beta}_{jk}^{\star} \,|\, \boldsymbol{\beta}_{jk}^{(t)}\right)$ **then**
    $\boldsymbol{\beta}_{jk}^{(t+1)} \leftarrow \boldsymbol{\beta}_{jk}^{\star}$
  else
    $\boldsymbol{\beta}_{jk}^{(t+1)} \leftarrow \boldsymbol{\beta}_{jk}^{(t)}$.
  end if
  Generate $\boldsymbol{\tau}_{jk}^{(t+1)}$ analogously.
Output: Next state $\boldsymbol{\beta}_{jk}^{(t+1)}$, $\boldsymbol{\tau}_{jk}^{(t+1)}$.

In each updating step in Algorithm A1 each $\boldsymbol{\tau}_{jk} = (\tau_{1jk}, \ldots, \tau_{L_{jk}jk})^\top$ is optimized one after the other using adaptive search intervals. Hence, the optimization problem is reduced to a one-dimensional search that is relative fast and straightforward to implement. The algorithm does not guarantee a global minimum given the goodness-of-fit criterion, however, the solution is at least close and serves as good starting points for full MCMC. Optimization speed can be further increased if for a given search interval only a grid of possible values for each τ_{ljk} is used.

The MCMC updating functions usually either accept or reject samples of the parameters and smoothing variances are sampled after $\boldsymbol{\beta}_{jk}$. In Algorithm A2b the general sampling scheme is shown. Note again the general structure of sampling Algorithm A2b, i.e., the proposal functions $q_{jk}(\cdot)$ generate parameter samples $\boldsymbol{\beta}_{jk}^{(t+1)}$, $\boldsymbol{\tau}_{jk}^{(t+1)}$ using (possibly) different sampling schemes like derivative-based Metropolis-Hastings and slice sampling, see Section 4.2.

4. Lego bricks

For computing parameter updates for either E1 or E2 using flexible partitioned updating systems like (7) and (8), the following "Lego bricks" are repeatedly used in Algorithm A1:

B1. The density $d_y(\mathbf{y}|\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K)$ and respective log-likelihood function $\ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})$,

B2. link functions $h_k(\cdot)$,

B3. model terms $f_{jk}(\cdot)$ and corresponding prior densities $p_{jk}(\boldsymbol{\beta}_{jk}; \boldsymbol{\tau}_{jk}, \boldsymbol{\alpha}_{jk})$.

Moreover, in this section we derive the details for updating Algorithms A2a and A2b, which usually require

B4. the derivatives of inverse link functions $h_k^{-1}(\cdot)$,

B5. the first order derivatives of the predictors $\frac{\partial \boldsymbol{\eta}_k}{\partial \boldsymbol{\beta}_{jk}}$,

B6. first order derivatives of the log-likelihood

- B6a. w.r.t. regression coefficients/parameters $\frac{\partial \ell(\boldsymbol{\beta};\mathbf{y},\mathbf{X})}{\partial \boldsymbol{\beta}_{jk}}$,
- B6b. w.r.t. predictors $\frac{\partial \ell(\boldsymbol{\beta};\mathbf{y},\mathbf{X})}{\partial \boldsymbol{\eta}_k}$,

B7. the second order derivatives of the log-likelihood

- B7a. w.r.t. regression coefficients/parameters $\frac{\partial^2 \ell(\boldsymbol{\beta};\mathbf{y},\mathbf{X})}{\partial \boldsymbol{\beta}_{jk} \partial \boldsymbol{\beta}_{jk}^\top}$,
- B7b. w.r.t. predictors $\frac{\partial^2 \ell(\boldsymbol{\beta};\mathbf{y},\mathbf{X})}{\partial \boldsymbol{\eta}_k \partial \boldsymbol{\eta}_k^\top}$,

B8. derivatives for log-priors, e.g., $\frac{\partial \log p_{jk}(\boldsymbol{\beta}_{jk};\boldsymbol{\tau}_{jk},\boldsymbol{\alpha}_{jk})}{\partial \boldsymbol{\beta}_{jk}}$.

Computationally, this leads to a "Lego" system and extending the toolbox can be done in different directions, e.g.: For a new response distribution, only building block B1, and possibly B6b and B7b are necessary, since in most cases B6a and B7a can be simplified when fragmenting with the chain rule. For a new model term B3 and B5 are needed. And for a new

link function B2 and B4. Then, the new building blocks are straightforward to combine with other previously available building blocks, moreover, most parts that are used for solving E1 can also be used for E2.

The remainder of this section is as follows. Details about commonly used prior densities in GAM-type models, building block B3, are provided in the next section. In Section 4.2 we derive the general parts that are needed for updating functions in Algorithm A2a and A2b, i.e., building blocks B6a, B6b, B7a and B7b. In Section 4.3 and 4.4 we briefly discuss model choice, Bayesian inference and prediction.

4.1. Model terms and priors

In the following we summarize commonly-used specifications for priors $p_{jk}(\cdot)$ used for estimating GAM-type models that can be used for building block B3. In addition, Table 1 provides an overview of model terms and prior structures.

Linear effects

For simple linear effects $f_{jk}(\cdot)$ a common choice for $p_{jk}(\cdot)$ is to use a non-informative (constant) flat prior. One of the simplest informative priors is a normal prior given by

$$p_{jk}(\boldsymbol{\beta}_{jk}; \boldsymbol{\tau}_{jk}, \boldsymbol{\alpha}_{jk}) \propto \exp\left(-\frac{1}{2}(\boldsymbol{\beta}_{jk} - \mathbf{m})^\top \mathbf{P}_{jk}(\boldsymbol{\tau}_{jk})(\boldsymbol{\beta}_{jk} - \mathbf{m})\right),$$

where $\boldsymbol{\tau}_{jk}$ are assumed to be fixed with $d_{\boldsymbol{\tau}_{jk}}(\cdot) = 1$ and $\boldsymbol{\alpha}_{jk} = \{\boldsymbol{\alpha}_{\boldsymbol{\beta}_{jk}} = \{\mathbf{m}\}\}$ with $\mathbf{m}$ as a prior mean for $\boldsymbol{\beta}_{jk}$. The matrix $\mathbf{P}_{jk}(\boldsymbol{\tau}_{jk})$ is a fixed prior precision matrix, e.g., $\mathbf{P}_{jk} = \text{diag}(\boldsymbol{\tau}_{jk})$. In a lot of applications a vague prior specification is used with $\mathbf{m} = \mathbf{0}$ and a large precision (see, e.g., Fahrmeir *et al.* 2013).

Nonlinear effects

If the nonlinear functions $f_{jk}(\cdot)$ in (1) are modeled using a basis function approach the usual choice of prior $p_{jk}(\cdot)$ is based on a multivariate normal kernel for $\boldsymbol{\beta}_{jk}$ given by

$$d_{\boldsymbol{\beta}_{jk}}(\boldsymbol{\beta}_{jk} | \, \boldsymbol{\tau}_{jk}, \boldsymbol{\alpha}_{\boldsymbol{\beta}_{jk}}) \propto |\mathbf{P}_{jk}(\boldsymbol{\tau}_{jk})|^{\frac{1}{2}} \exp\left(-\frac{1}{2}\boldsymbol{\beta}_{jk}^\top \mathbf{P}_{jk}(\boldsymbol{\tau}_{jk})\boldsymbol{\beta}_{jk}\right). \tag{9}$$

Here, the precision matrix $\mathbf{P}_{jk}(\boldsymbol{\tau}_{jk})$ is derived from prespecified so-called penalty matrices $\boldsymbol{\alpha}_{\boldsymbol{\beta}_{jk}} = \{\mathbf{K}_{1jk}, \ldots, \mathbf{K}_{Ljk}\}$, e.g., for tensor product smooths the precision matrix is $\mathbf{P}_{jk}(\boldsymbol{\tau}_{jk}) = \sum_{l=1}^{L_{jk}} \frac{1}{\tau_{ljk}} \mathbf{K}_{ljk}$. Note that $\mathbf{P}_{jk}(\boldsymbol{\tau}_{jk})$ is often not of full rank, therefore, $d_{\boldsymbol{\beta}_{jk}}(\cdot)$ is partially improper. The variances τ_{ljk} account for the amount of smoothness (regularization) of the function and can be interpreted as the inverse smoothing parameters in the frequentist approach. A common choice for the prior for $\boldsymbol{\tau}_{jk}$ is based on inverse gamma distributions for each $\boldsymbol{\tau}_{jk} = (\tau_{1jk}, \ldots, \tau_{L_{jk}jk})^\top$

$$d_{\boldsymbol{\tau}_{jk}}(\boldsymbol{\tau}_{jk} | \, \boldsymbol{\alpha}_{\boldsymbol{\tau}_{jk}}) = \prod_{l=1}^{L_{jk}} \frac{b_{ljk}^{a_{ljk}}}{\Gamma(a_{ljk})} \tau_{ljk}^{-(a_{ljk}+1)} \exp(-b_{ljk}/\tau_{ljk}), \tag{10}$$

with fixed parameters $\boldsymbol{\alpha}_{\boldsymbol{\tau}_{jk}} = \{\mathbf{a}_{jk}, \mathbf{b}_{jk}\}$. Small values for $\mathbf{a}_{jk}$ and $\mathbf{b}_{jk}$ correspond to approximate flat priors for $\log(\tau_{ljk})$. Setting $\mathbf{b}_{jk} = \mathbf{0}$ and $\mathbf{a}_{jk} = -\mathbf{1}$ or $\mathbf{a}_{jk} = -1/2 \cdot \mathbf{1}$ yields flat

Covariates	Effect types $f_{jk}(\mathbf{x};\boldsymbol{\beta}_{jk})$	Prior densities $p_{jk}(\boldsymbol{\beta}_{jk};\boldsymbol{\tau}_{jk},\boldsymbol{\alpha}_{jk})$ $d_{\boldsymbol{\beta}_{jk}}(\boldsymbol{\beta}_{jk}\vert\,\boldsymbol{\tau}_{jk};\boldsymbol{\alpha}_{\boldsymbol{\beta}_{jk}})$	$d_{\tau_{ljk}}(\tau_{ljk}\vert\,\boldsymbol{\alpha}_{\tau_{ljk}})$
Scalar covariates	Intercept β Linear effect $x\cdot\beta$ Linear interaction $x_1\cdot x_2\cdot\beta$	$\propto$ constant $\propto \exp\left(-\frac{1}{2}(\boldsymbol{\beta}-\mathbf{m})^\top\mathbf{P}(\boldsymbol{\tau})(\boldsymbol{\beta}-\mathbf{m})\right)$	$\emptyset$
Grouping variable s	Smooth effect $f(x)$ Varying coefficient $f(x_2)\cdot x_1$ Smooth effect $f(x_1,\ldots,x_L)$ Random intercept β_s Spatial effect $f(s)$	$\propto \lvert\mathbf{P}(\boldsymbol{\tau})\rvert^{\frac{1}{2}}\exp\left(-\frac{1}{2}\boldsymbol{\beta}^\top\mathbf{P}(\boldsymbol{\tau})\boldsymbol{\beta}\right)$	IG $\propto \tau^{-(a+1)}\exp(-b/\tau)$ HC $\propto (1+\tau/a^2)^{-1}(\tau/a^2)^{-1/2}$
Grouping and scalar, time variable t	Random slope $x\cdot\beta_s$ Space-time effect $f(s,t)$ Functional random intercept $f_s(t)$		SD $\propto (\tau/\sqrt{\tau})^{-1/2}\exp(-(\tau/a)^{1/2})$ HN $\propto \tau^{1/2-1}\exp(-\tau/(2a^2))$

Table 1: Commonly used "Lego bricks", building block B3, for model terms in BAMLSS. Priors for linear effects assume that the precision matrix $\mathbf{P}(\boldsymbol{\tau})$ is fixed. For smooth effects, prior densities are: inverse gamma (IG), half-Cauchy (HC), scale-dependent (SD) and half-normal (HN).

priors for τ_{ljk} and $\tau_{ljk}^{0.5}$, respectively. However, the inverse gamma prior might be problematic if τ_{ljk} is close to zero, since the results a very sensitive to the choice of $\mathbf{a}_{jk}$ and $\mathbf{b}_{jk}$. Therefore, Gelman (2006) proposes to use the half-Cauchy prior

$$d_{\boldsymbol{\tau}_{jk}}(\boldsymbol{\tau}_{jk}|\,\boldsymbol{\alpha}_{\boldsymbol{\tau}_{jk}}) = \prod_{l=1}^{L_{jk}} \frac{2A_{ljk}}{\pi(\tau_{ljk} + A_{ljk}^2)}, \quad A_{ljk} > 0,$$

with hyper-parameters $\boldsymbol{\alpha}_{\boldsymbol{\tau}_{jk}} = \{\mathbf{A}_{jk}\}$. For $A_{ljk} \to \infty$ the priors are uniform, hence large values (e.g., $A_{ljk} = 25$) result in weakly informative priors. A desirable property of the half-Cauchy is that for $\tau_{ljk} = 0$ the density is a nonzero constant, whereas the density of the inverse gamma for $\tau_{ljk} \to 0$ vanishes (see also Polson and Scott 2012). Another question is the actual choice of hyper-parameters. A recent suggestion reducing this issue to the choice of a scale parameter that is directly related to the functions $f_{jk}(\cdot)$ (and thus much better interpretable and accessible for the user) is given in Klein and Kneib (2016a) for several different hyper-priors for τ_{ljk}, such as resulting priors from half-Cauchy, half-normal or uniform priors for $\tau_{ljk}^{0.5}$ or resulting penalized complexity priors (Simpson, Rue, Martins, Riebler, and Sørbye 2017), so-called scale-dependent priors.

Multilevel effects

In numerous applications geographical information and spatial covariates are given at different resolutions. For example, spatial data that is measured within different regions, for which additional regional covariates are available. Whenever there is such a nested structure in the data, it is possible to model the complex (spatial) heterogeneity effects using a compound prior

$$\boldsymbol{\beta}_{jk} = \tilde{\boldsymbol{\eta}}_{jk}(\mathbf{x}; \tilde{\boldsymbol{\beta}}_{jk}) + \boldsymbol{\varepsilon}_{jk},$$

where $\boldsymbol{\varepsilon}_{jk} \sim \mathcal{N}(\mathbf{0}, \tilde{\tau}_{jk}\mathbf{I})$ is a vector of i.i.d. Gaussian random effects and $\tilde{\boldsymbol{\eta}}_{jk}(\mathbf{x}; \tilde{\boldsymbol{\beta}}_{jk})$ represents a full predictor of nested covariates, e.g., including a discrete regional spatial effect. This way, potential costly operations in updating Algorithm A2a and A2b can be avoided since the number of observations in $\tilde{\boldsymbol{\eta}}_{jk}(\mathbf{x}; \tilde{\boldsymbol{\beta}}_{jk})$ is equal to the number of coefficients in $\boldsymbol{\beta}_{jk}$, which is usually much smaller than the actual number of observations n. Moreover, the full conditionals (see also Section 4.2) for $\tilde{\boldsymbol{\beta}}_{jk}$ are Gaussian regardless of the response distribution and leads to highly efficient estimation algorithms, see Lang, Umlauf, Wechselberger, Harttgen, and Kneib (2014).

4.2. Model fitting

The construction of suitable updating functions $U_{jk}(\cdot)$ for solving problem E1 and E2 can be carried out in many ways. In this respect, note again that algorithm A1 is very general, i.e., does not restrict to a specific iterative procedure. In the following we describe commonly used quantities that can be used for estimation of BAMLSS. Moreover, this section highlights the "Lego" system character described above, that arises when using gradient-based updating schemes for E1 and E2. More precisely, we first describe posterior mode updating as used within Algorithm A2a, before we introduce several MCMC sampling schemes that can be employed in the updating Algorithm A2b.

Posterior mode

The mode of the posterior distribution is the mode of the log-posterior (4) given by

$$\text{Mode}(\boldsymbol{\beta}, \boldsymbol{\tau} | \, \mathbf{y}, \mathbf{X}, \boldsymbol{\alpha}) = \arg\max_{\boldsymbol{\beta}, \boldsymbol{\tau}} \; \log \pi(\boldsymbol{\beta}, \boldsymbol{\tau}; \mathbf{y}, \mathbf{X}, \boldsymbol{\alpha})$$

and is equivalent to the maximum likelihood estimator

$$\text{ML}(\boldsymbol{\beta} | \, \mathbf{y}, \mathbf{X}) = \arg\max_{\boldsymbol{\beta}} \, \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})$$

when assigning flat (constant) priors to $\boldsymbol{\beta}_{jk}$ for $j = 1, \ldots, J_k$, $k = 1, \ldots, K$. For models involving shrinkage priors, e.g., in GAM-type models given by (9), the posterior mode is equivalent to a penalized maximum likelihood estimator for fixed parameters $\boldsymbol{\tau}_{jk}$ and prior densities $d_{\boldsymbol{\tau}_{jk}}(\cdot) \propto$ constant. Moreover, the structure of the log-posterior (4) usually prohibits estimation of $\boldsymbol{\tau}_{jk}$ through maximization and the estimator $\hat{\boldsymbol{\tau}}_{jk}$ is commonly derived by additionally minimizing an information criterion such as the Akaike information criterion (AIC) or the Bayesian information criterion (BIC). See also Algorithm A2a for an adaptive stepwise approach for estimation of $\boldsymbol{\tau}_{jk}$ (see also Rigby and Stasinopoulos 2005 Appendix A.2. for a more detailed discussion on smoothing parameter estimation). In Section 4.3, we briefly discuss details on the computation of information criteria with equivalent degrees of freedom.

For developing general updating functions we begin with describing posterior mode estimation for the case of fixed parameters $\boldsymbol{\tau}_{jk}$, because these updating functions form the basis of estimation algorithms for $\boldsymbol{\tau}_{jk}$. Estimation of $\boldsymbol{\beta} = (\boldsymbol{\beta}_1^\top, \ldots, \boldsymbol{\beta}_K^\top)^\top$ requires solving equations $\partial(\log \pi(\boldsymbol{\beta}, \boldsymbol{\tau}; \mathbf{y}, \mathbf{X}, \boldsymbol{\alpha}))/\partial\boldsymbol{\beta} = \mathbf{0}$. A particularly convenient updating function (6) for maximization of (4) is a Newton-Raphson type updating

$$\boldsymbol{\beta}^{(t+1)} = U(\boldsymbol{\beta}^{(t)}, \cdot) = \boldsymbol{\beta}^{(t)} - \mathbf{H}\left(\boldsymbol{\beta}^{(t)}\right)^{-1} \mathbf{s}\left(\boldsymbol{\beta}^{(t)}\right) \tag{11}$$

with score vector

$$\mathbf{s}(\boldsymbol{\beta}) = \frac{\partial \log \pi(\boldsymbol{\beta}, \boldsymbol{\tau}; \mathbf{y}, \mathbf{X}, \boldsymbol{\alpha})}{\partial \boldsymbol{\beta}} = \frac{\partial \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})}{\partial \boldsymbol{\beta}} + \sum_{k=1}^{K} \sum_{j=1}^{J_k} \left[\frac{\partial \log p_{jk}(\boldsymbol{\beta}_{jk}; \boldsymbol{\tau}_{jk}, \boldsymbol{\alpha}_{jk})}{\partial \boldsymbol{\beta}} \right].$$

and Hessian matrix $\mathbf{H}(\boldsymbol{\beta})$ with components

$$\mathbf{H}_{ks}(\boldsymbol{\beta}) = \frac{\partial \mathbf{s}(\boldsymbol{\beta}_k)}{\partial \boldsymbol{\beta}_s^\top} = \frac{\partial^2 \log \pi(\boldsymbol{\beta}, \boldsymbol{\tau}; \mathbf{y}, \mathbf{X}, \boldsymbol{\alpha})}{\partial \boldsymbol{\beta}_k \partial \boldsymbol{\beta}_s^\top},$$

for $k = 1, \ldots, K$ and $s = 1, \ldots, K$. By chain rule, the part of the score vector involving the derivatives of the log-likelihood for the k-th parameter can be further decomposed to

$$\frac{\partial \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})}{\partial \boldsymbol{\beta}_k} = \frac{\partial \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})}{\partial \boldsymbol{\eta}_k} \frac{\partial \boldsymbol{\eta}_k}{\partial \boldsymbol{\beta}_k} = \frac{\partial \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})}{\partial \boldsymbol{\theta}_k} \frac{\partial \boldsymbol{\theta}_k}{\partial \boldsymbol{\eta}_k} \frac{\partial \boldsymbol{\eta}_k}{\partial \boldsymbol{\beta}_k},$$

including the derivatives of the log-likelihood with respect to $\boldsymbol{\eta}_k$ and $\boldsymbol{\theta}_k$, building block B6a, the derivative of the inverse link functions, component B4, and the derivative of the predictor

$\boldsymbol{\eta}_k$ with respect to coefficients $\boldsymbol{\beta}_k$, B5. Again by chain rule, the components of $\mathbf{H}_{ks}$ including $\ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})$, building block B7a, can be written as

$$\mathbf{J}_{ks}(\boldsymbol{\beta}) = \frac{\partial^2 \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})}{\partial \boldsymbol{\beta}_k \partial \boldsymbol{\beta}_s^\top} = \left(\frac{\partial \boldsymbol{\eta}_s}{\partial \boldsymbol{\beta}_s}\right)^\top \frac{\partial^2 \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})}{\partial \boldsymbol{\eta}_k \partial \boldsymbol{\eta}_s^\top} \frac{\partial \boldsymbol{\eta}_k}{\partial \boldsymbol{\beta}_k} \underbrace{+ \frac{\partial \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})}{\partial \boldsymbol{\eta}_k} \frac{\partial^2 \boldsymbol{\eta}_k}{\partial^2 \boldsymbol{\beta}_k}}_{\text{if } k=s}, \tag{12}$$

yielding a decomposition of building blocks B7b and B5. The second term on the right hand side cancels out if all functions (2) can be written as a matrix product of a design matrix and coefficients, e.g., when using a basis function approach. Within the first term, the second derivatives of the log-likelihood involving the predictors can be written as

$$\frac{\partial^2 \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})}{\partial \boldsymbol{\eta}_k \partial \boldsymbol{\eta}_s^\top} = \frac{\partial \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})}{\partial \boldsymbol{\theta}_k} \frac{\partial^2 \boldsymbol{\theta}_k}{\partial \boldsymbol{\eta}_k \partial \boldsymbol{\eta}_s^\top} + \frac{\partial^2 \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})}{\partial \boldsymbol{\theta}_k \partial \boldsymbol{\theta}_s^\top} \frac{\partial \boldsymbol{\theta}_k}{\partial \boldsymbol{\eta}_k} \frac{\partial \boldsymbol{\theta}_s}{\partial \boldsymbol{\eta}_s} \tag{13}$$

involving the second derivatives of the link functions.

Using a k-partitioned updating scheme as presented in (7) updating functions $U_k(\cdot)$ are given by

$$\boldsymbol{\beta}_k^{(t+1)} = U_k(\boldsymbol{\beta}_k^{(t)}, \cdot) = \boldsymbol{\beta}_k^{(t)} - \mathbf{H}_{kk}\left(\boldsymbol{\beta}_k^{(t)}\right)^{-1} \mathbf{s}\left(\boldsymbol{\beta}_k^{(t)}\right). \tag{14}$$

Assuming model terms (2) that can be written as a matrix product of a design matrix and coefficients the Hessian matrix in (14) is given by

$$\mathbf{H}_{kk}\left(\boldsymbol{\beta}_k^{(t)}\right) = \begin{pmatrix} \mathbf{X}_{1k}^\top \mathbf{W}_{kk} \mathbf{X}_{1k} + \mathbf{G}_{1k}(\boldsymbol{\tau}_{1k}) & \cdots & \mathbf{X}_{1k}^\top \mathbf{W}_{kk} \mathbf{X}_{J_k k} \\ \vdots & \ddots & \vdots \\ \mathbf{X}_{J_k k}^\top \mathbf{W}_{kk} \mathbf{X}_{1k} & \cdots & \mathbf{X}_{J_k k}^\top \mathbf{W}_{kk} \mathbf{X}_{J_k k} + \mathbf{G}_{J_k k}(\boldsymbol{\tau}_{J_k k}) \end{pmatrix}^{(t)},$$

with diagonal weight matrix $\mathbf{W}_{kk} = -\text{diag}(\partial^2 \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X}) / \partial \boldsymbol{\eta}_k \partial \boldsymbol{\eta}_k^\top)$ and matrices forming building block B8

$$\mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}) = \frac{\partial^2 \log p_{jk}(\boldsymbol{\beta}_{jk}; \boldsymbol{\tau}_{jk}, \boldsymbol{\alpha}_{jk})}{\partial \boldsymbol{\beta}_{jk} \partial \boldsymbol{\beta}_{jk}^\top}. \tag{15}$$

Here, we want to emphasize that the influence of these prior derivatives matrices is usually controlled by $\boldsymbol{\tau}_{jk}$, however, note once again that the $\boldsymbol{\tau}_{jk}$ are held fixed for the moment and usually estimation cannot be done with maximization of the log-posterior (see also Section 4.3). Typically, using a linear basis function representation of functions $f_{jk}(\cdot)$ and priors based on multivariate normal kernels (9) matrices $\mathbf{G}_{jk}(\boldsymbol{\tau}_{jk})$ are a simple product of smoothing variances and penalty matrices, e.g., with only one smoothing variance building block B8 becomes $\mathbf{G}_{jk}(\tau_{jk}) = \tau_{jk}^{-1} \mathbf{K}_{jk}$ with corresponding penalty matrix $\mathbf{K}_{jk}$.

Similarly, the score vector is

$$\mathbf{s}\left(\boldsymbol{\beta}_k^{(t)}\right) = \begin{pmatrix} \mathbf{X}_{1k}^\top \mathbf{u}_k^{(t)} - \mathbf{G}_{1k}(\boldsymbol{\tau}_{1k}) \boldsymbol{\beta}_{1k}^{(t)} \\ \vdots \\ \mathbf{X}_{J_k k}^\top \mathbf{u}_k^{(t)} - \mathbf{G}_{J_k k}(\boldsymbol{\tau}_{J_k k}) \boldsymbol{\beta}_{J_k k}^{(t)} \end{pmatrix}$$

and derivatives $\mathbf{u}_k = \partial \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X}) / \partial \boldsymbol{\eta}_k$. Focusing on the j-th row of (14) leads to single model term updating functions $U_{jk}(\cdot)$ as presented in algorithm (8). The updates are based on an iteratively weighted least squares scheme given by

$$\boldsymbol{\beta}_{jk}^{(t+1)} = U_{jk}(\boldsymbol{\beta}_{jk}^{(t)}, \cdot) = (\mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}))^{-1} \mathbf{X}_{jk}^\top \mathbf{W}_{kk} (\mathbf{z}_k - \boldsymbol{\eta}_{k,-j}^{(t+1)}) \tag{16}$$

with working observations $\mathbf{z}_k = \boldsymbol{\eta}_k^{(t)} + \mathbf{W}_{kk}^{-1}\mathbf{u}_k^{(t)}$ (in Appendix A the detailed derivations are presented), i.e., the algorithm only requires building blocks B6b, B7b and B8. Hence, this leads to a backfitting algorithm and cycling through (16) for terms $j = 1, \ldots, J_k$ and parameters $k = 1, \ldots, K$ approximates a single Newton-Raphson step in (11), since cross derivatives are not incorporated in the updating scheme. Note that this yields the ingredients of the *RS*-algorithm developed in Rigby and Stasinopoulos (2005) Appendix B.2. The updating scheme (16) can be further generalized to

$$\boldsymbol{\beta}_{jk}^{(t+1)} = U_{jk}\left(\boldsymbol{\beta}_{jk}^{(t)}, \mathbf{z}_k - \boldsymbol{\eta}_{k,-j}^{(t+1)}, \cdot\right)$$

i.e., theoretically any updating function applied on the "partial residuals" $\mathbf{z}_k - \boldsymbol{\eta}_{k,-j}^{(t+1)}$ can be used. Note also that this result is equivalent to updating function

$$\begin{aligned}
\boldsymbol{\beta}_{jk}^{(t+1)} &= U_{jk}(\boldsymbol{\beta}_{jk}^{(t)}, \cdot) \\
&= \boldsymbol{\beta}_{jk}^{(t)} - \mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)^{-1}\mathbf{s}\left(\boldsymbol{\beta}_{jk}^{(t)}\right) \\
&= \boldsymbol{\beta}_{jk}^{(t)} - \left[\mathbf{J}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right) + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk})\right]^{-1}\mathbf{s}\left(\boldsymbol{\beta}_{jk}^{(t)}\right),
\end{aligned} \tag{17}$$

where matrix $\mathbf{J}_{kk}(\cdot)$ is the derivative matrix given in (12), involving building blocks B6a, B7a and B8.

For optimization, different strategies of the backfitting algorithm (16) can be applied. One alternative is a complete inner backfitting algorithm for each parameter k, i.e., the backfitting procedure updates $\boldsymbol{\beta}_{jk}$, for $j = 1, \ldots, J_k$ until convergence, afterwards updates for parameters for the next k are calculated again by a complete inner backfitting algorithm, and so forth (see also Rigby and Stasinopoulos 2005).

Note that for numerical reasons it is oftentimes better to replace the Hessian by the expected Fisher information with weights $\mathbf{W}_{kk} = -\text{diag}(E(\partial^2\ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})/\partial\boldsymbol{\eta}_k\partial\boldsymbol{\eta}_k^\top))$, see Klein *et al.* (2015b). Moreover, to achieve convergence, algorithms for posterior mode usually initialize the parameter vectors $\boldsymbol{\theta}_k$. Then, after one complete inner backfitting iteration the algorithm can proceed in a full zigzag fashion or again with inner iterations. For all updating schemes it might also be appropriate to vary the updating step length of parameter updates (half-stepping), possibly in each iteration. This is relatively straightforward to implement, because step length optimization is a one-dimensional problem, i.e., for each model term finding the step length that improves the log-posterior most.

Posterior mean

The mean of the posterior distribution is

$$E(\boldsymbol{\beta}, \boldsymbol{\tau} | \mathbf{y}, \mathbf{X}, \boldsymbol{\alpha}) = \int \begin{pmatrix}\boldsymbol{\beta}\\ \boldsymbol{\tau}\end{pmatrix} \pi(\boldsymbol{\beta}, \boldsymbol{\tau}; \mathbf{y}, \mathbf{X}, \boldsymbol{\alpha}) d \begin{pmatrix}\boldsymbol{\beta}\\ \boldsymbol{\tau}\end{pmatrix}.$$

Clearly, the problem in deriving the expectation, and other quantities like the posterior median, relies on the computation of usually high-dimensional integrals, which can be rarely solved analytically and thus need to be approximated by numerical techniques.

MCMC simulation is commonly used in such situations as it provides an extensible framework that can adapt to almost any type of problem. In the following we summarize sampling

techniques that are especially well-suited within the BAMLSS framework, i.e., techniques that can be used for a highly modular and extensible system. In this context we describe sampling functions for the updating scheme presented in (8), i.e., the functions $U_{jk}(\cdot)$ now generate the next step in a Markov chain.

Note that for some models there exist full conditionals that can be derived in closed form from the log-posterior (4). However, we especially focus on situations were this is not generally the case. MCMC samples for the regression coefficients $\boldsymbol{\beta}_{jk}$ can be derived by each of the following methods:

- *Random-walk Metropolis*:
  Probably the most important algorithm, because of its generality and ease of implementation, is random-walk Metropolis. The sampler proceeds by drawing a candidate $\boldsymbol{\beta}_{jk}^{\star}$ from a symmetric jumping distribution $q(\boldsymbol{\beta}_{jk}^{\star}|\,\boldsymbol{\beta}_{jk}^{(t)})$, the candidate is then accepted as the new state of the Markov chain with probability

  $$\alpha\left(\boldsymbol{\beta}_{jk}^{\star}|\,\boldsymbol{\beta}_{jk}^{(t)}\right) = \min\left[\frac{\pi(\boldsymbol{\beta}_{jk}^{\star}|\,\cdot)}{\pi(\boldsymbol{\beta}_{jk}^{(t)}|\,\cdot)}, 1\right]$$

  with the log-posterior $\pi(\boldsymbol{\beta}_{jk}|\,\cdot)$ evaluated at the proposed and current value. Commonly, the jumping distribution is a normal distribution $\mathcal{N}(\boldsymbol{\beta}_{jk}^{(t)}, \boldsymbol{\Sigma}_{jk})$ centered at the current iterate and fixed covariance matrix. Although this algorithm is theoretically working for any distribution, the actual sampling performance depends heavily on starting values and the scaling of $\boldsymbol{\Sigma}_{jk}$. Therefore, numerous methods that try to optimize the behavior of the Markov chain in an adaptive phase (burnin phase) have been developed. In the seminal paper of Gelman, Roberts, and Gilks (1996), strategies that optimize the acceptance rate to roughly 1/4 are suggested to obtain a good mixing (see also Roberts and Rosenthal 2009). Similarly, within the presented modeling framework and a basis function approach with multivariate normal prior (9), a convenient way is to set $\boldsymbol{\Sigma}_{jk} = \sigma_{jk}\mathbf{P}_{jk}(\boldsymbol{\tau}_{jk})^{-1}$ and optimize σ_{jk} to the desired properties in the adaptive phase.

- *Derivative-based Metropolis-Hastings*:
  A commonly used alternative for the covariance matrix of the jumping distribution $\mathcal{N}(\boldsymbol{\beta}_{jk}^{(t)}, \boldsymbol{\Sigma}_{jk})$ is to use the local curvature information

  $$\boldsymbol{\Sigma}_{jk} = -\left(\frac{\partial^2 \pi(\boldsymbol{\vartheta}; \mathbf{y}, \mathbf{X})}{\partial \boldsymbol{\beta}_{jk} \boldsymbol{\beta}_{jk}^{\top}}\right)^{-1},$$

  or its expectation, computed at the posterior mode estimate $\hat{\boldsymbol{\beta}}_{jk}$, requiring building blocks B7a and B8. However, fixing $\boldsymbol{\Sigma}_{jk}$ during MCMC simulation might still lead to undesired behavior of the Markov chain especially when parameter samples move into regions with low probability mass of the posterior distribution. A solution with good mixing properties is to construct approximate full conditionals $\pi(\boldsymbol{\beta}_{jk}|\cdot)$ that are based on a second order Taylor series expansion of the log-posterior centered at the last state (Gamerman 1997; Fahrmeir *et al.* 2004; Brezger and Lang 2006; Klein and Kneib 2016b). The resulting proposal density is multivariate normal (see Appendix B for a

detailed derivation) with precision matrix

$$\left(\boldsymbol{\Sigma}_{jk}^{(t)}\right)^{-1} = -\mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)$$

and mean

$$\begin{aligned}\boldsymbol{\mu}_{jk}^{(t)} &= \boldsymbol{\Sigma}_{jk}^{(t)}\left[s\left(\boldsymbol{\beta}_{jk}^{(t)}\right) - \mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)\boldsymbol{\beta}_{jk}^{(t)}\right] \\ &= \boldsymbol{\beta}_{jk}^{(t)} - \mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)^{-1}\mathbf{s}\left(\boldsymbol{\beta}_{jk}^{(t)}\right) \\ &= \boldsymbol{\beta}_{jk}^{(t)} - \left[\mathbf{J}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right) + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk})\right]^{-1}\mathbf{s}\left(\boldsymbol{\beta}_{jk}^{(t)}\right),\end{aligned}$$

which is equivalent to the updating function given in (17) and can again be build using blocks B7a and B8. Hence, the mean is simply one Newton or Fisher scoring iteration towards the posterior mode at the current step. The proposal density for $\boldsymbol{\beta}_{jk}$ then is $q(\boldsymbol{\beta}_{jk}^{\star}|\boldsymbol{\beta}_{jk}^{(t)}) = \mathcal{N}(\boldsymbol{\mu}_{jk}^{(t)}, \boldsymbol{\Sigma}_{jk}^{(t)})$ and the acceptance probability of the candidate is computed by

$$\alpha\left(\boldsymbol{\beta}_{jk}^{\star}|\,\boldsymbol{\beta}_{jk}^{(t)}\right) = \min\left[\frac{\pi(\boldsymbol{\beta}_{jk}^{\star}|\,\cdot)q(\boldsymbol{\beta}_{jk}^{(t)}|\,\boldsymbol{\beta}_{jk}^{\star})}{\pi(\boldsymbol{\beta}_{jk}^{(t)}|\,\cdot)q(\boldsymbol{\beta}_{jk}^{\star}|\,\boldsymbol{\beta}_{jk}^{(t)})}, 1\right].$$

Again, assuming a basis function approach for functions $f_{jk}(\cdot)$ the precision matrix is

$$\left(\boldsymbol{\Sigma}_{jk}^{(t)}\right)^{-1} = \mathbf{X}_{jk}^{\top}\mathbf{W}_{kk}\mathbf{X}_{jk} + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}),$$

with weights $\mathbf{W}_{kk} = -\text{diag}(\partial^2\ell(\boldsymbol{\beta};\mathbf{y},\mathbf{X})/\partial\boldsymbol{\eta}_k\partial\boldsymbol{\eta}_k^{\top})$, or the corresponding expectation, as in posterior mode updating using building blocks B7b and B8. The mean can be written as

$$\boldsymbol{\mu}_{jk}^{(t)} = \boldsymbol{\Sigma}_{jk}^{(t)}\mathbf{X}_{jk}^{\top}\mathbf{W}_{kk}\left(\mathbf{z}_k - \boldsymbol{\eta}_{k,-j}^{(t)}\right)$$

with working observations $\mathbf{z}_k = \boldsymbol{\eta}_k^{(t)} + \mathbf{W}_{kk}^{-1}\mathbf{u}_k^{(t)}$ (see again Appendix B for a detailed derivation). Note again, the computation of the mean is equivalent to a full Newton step as given in updating function (17), or Fisher scoring when using $-E(\partial^2\ell(\boldsymbol{\beta};\mathbf{y},\mathbf{X})/\partial\boldsymbol{\eta}_k\partial\boldsymbol{\eta}_k^{\top})$, in each iteration of the MCMC sampler using iteratively weighted least squares (IWLS). If the computation of the weights $\mathbf{W}_{kk}$ is expensive, one simple strategy is to update $\mathbf{W}_{kk}$ only after samples of all parameters of $\boldsymbol{\theta}_k$ are drawn.

- *Slice sampling*:
  Slice sampling (Neal 2003) is a gradient free MCMC sampling scheme that produces samples with 100% acceptance rate. Therefore, and because of the simplicity of the algorithm, slice sampling is especially useful for automated general purpose MCMC implementations that allow for sampling from many distributions. The basic slice sampling algorithm samples univariate directly under the plot of the log-posterior (4). Updates for the i-th parameter in $\boldsymbol{\beta}_{jk}$ are generated by:

  1. Sample $h \sim \mathcal{U}(0, \pi(\beta_{ijk}^{(t)}|\,\cdot))$.
  2. Sample $\beta_{ijk}^{(t+1)} \sim \mathcal{U}(S)$ from the horizontal slice $S = \{\beta_{ijk} : h < \pi(\beta_{ijk}|\,\cdot)\}$.

The full conditional $\pi(\boldsymbol{\tau}_{jk}|\cdot)$ for smoothing variances is commonly constructed using priors for $\boldsymbol{\tau}_{jk}$ that lead to known distributions, i.e., simple Gibbs sampling is possible. E.g., this is the case when using a basis function approach and only one smoothing variance τ_{jk} is assigned. Then, by using an inverse gamma prior (10) for τ_{jk} in combination with the normal prior (9) for $\boldsymbol{\beta}_{jk}$ the full-conditional $\pi(\tau_{jk}|\cdot)$ is again an inverse gamma distribution with

$$\tilde{a}_{jk} = \frac{1}{2} rk(\mathbf{K}_{jk}) + a_{jk}, \qquad \tilde{b}_{jk} = \frac{1}{2}(\boldsymbol{\beta}_{jk}^{\star})^{\top}\mathbf{K}_{jk}\boldsymbol{\beta}_{jk}^{\star} + b_{jk},$$

and matrix $\mathbf{K}_{jk}$ is again a penalty matrix that depends on the type of model term. As mentioned in Section 4.1, other priors than the inverse gamma might be desirable. Then, Metropolis-Hastings steps also for the variances can be constructed, see Klein and Kneib (2016a) for details. If a simple Gibbs sampling step cannot be derived, e.g., for multi-dimensional tensor product splines, another strategy is to use slice sampling, since the number of smoothing variances is usually not very large the computational burden does most of the times not exceed possible benefits.

4.3. Model choice

In the context of BAMLSS, model choice is usually based on the full response distribution. In the following commonly used methods used for model choice and variable selection are outlined.

Diagnostics

Quantile residuals defined as $\hat{r}_i = \Phi^{-1}(\mathcal{F}(y_i|\,\hat{\boldsymbol{\theta}}_i))$ with the inverse cumulative distribution function of a standard normal distribution Φ^{-1} and $\mathcal{F}(\cdot)$ denoting the cumulative distribution function (CDF) of the modeled distribution $\mathcal{D}(\cdot)$ with estimated parameters $\hat{\boldsymbol{\theta}}_i = (\hat{\theta}_{i1}, \ldots, \hat{\theta}_{iK})^{\top}$ plugged in, should at least approximately be standard normally distributed if the correct model has been specified (Dunn and Smyth 1996; Klein *et al.* 2015b). Resulting residuals can be assessed graphically in terms of quantile-quantile-plots. Strong deviations from the diagonal line are then a sign for an inappropriate model fit. Instead of looking at residuals one can use the probability integral transform (PIT, Gneiting, Balabdaoui, and Raftery 2007) which considers $u_i = \mathcal{F}(y_i|\,\hat{\boldsymbol{\theta}}_i)$ without applying the inverse standard normal CDF. If the estimated model is a good approximation to the true data generating process, the u_i will then approximately follow a uniform distribution on $[0, 1]$. Graphically, histograms of the u_i can be used for this purpose.

Smoothing variances with posterior mode

As already mentioned in Section 4.2, depending on the structure of the priors (5) parameters $\boldsymbol{\tau}_{jk}$ cannot be estimated by maximization of the log-posterior (4). For example, this is the case for GAM-type models in combination with priors based on multivariate normal kernels (9) where $\boldsymbol{\tau}_{jk}$ represents smoothing variances.

Therefore, goodness-of-fit criteria like the Akaike information criterion (AIC), or the corrected AIC, as well as the Bayesian information criterion (BIC), amongst others, are commonly used for selecting the smoothing variances $\boldsymbol{\tau}_{jk}$. These criteria try to penalize overly complex models, i.e., are trying to prevent over-fitting. For models using a basis function approach, estimating model complexity using (possibly) nonlinear functions is based on the so-called

equivalent degrees of freedom (EDF). For each model component the EDF used to estimate the function are calculated by

$$\text{edf}_{jk}(\boldsymbol{\tau}_{jk}) := \text{trace}\left[\mathbf{J}_{kk}(\boldsymbol{\beta}_{jk})(\mathbf{J}_{kk}(\boldsymbol{\beta}_{jk}) + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}))^{-1}\right],$$

where $\mathbf{J}_{kk}(\cdot)$ is the derivative matrix given in (12) and matrix $\mathbf{G}_{jk}(\boldsymbol{\tau}_{jk})$ is the prior derivative matrix as given in (15). The total degrees of freedom used to fit the model are then estimated by $\sum_k \sum_j \text{edf}_{jk}(\boldsymbol{\tau}_{jk})$. Note that the definition of EDF here is slightly more general and is usually defined as the trace of the smoother matrix (see, e.g., Hastie and Tibshirani 1990) and can be applied even for more complex likelihood structures, e.g., in a flexible Cox model (Hofner 2008).

Instead of global optimization of smoothing variances, a fast strategy is the adaptive stepwise selection approach presented in Algorithm A2a.

Variable selection with posterior mean

The deviance information criterion (DIC) can be used for model choice and variable selection in Bayesian inference. It is easily be computed from the MCMC output without requiring additional computational effort. If $\boldsymbol{\beta}^{(1)}, \ldots, \boldsymbol{\beta}^{(T)}$ is a MCMC sample from the posterior for the complete parameter vector $\boldsymbol{\beta}$, the DIC is given by $\overline{D(\boldsymbol{\beta})}+\text{pd} = 2\overline{D(\boldsymbol{\beta})} - D(\overline{\boldsymbol{\beta}}) = \frac{2}{T}\sum D(\boldsymbol{\beta}^{(t)}) - D(\frac{1}{T}\sum \boldsymbol{\beta}^{(t)})$ where $D(\boldsymbol{\beta}) = -2 \cdot \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})$ is the model deviance and $\text{pd} = \overline{D(\boldsymbol{\beta})} - D(\overline{\boldsymbol{\beta}})$ is an effective parameter count.

4.4. Inference and prediction

Under suitable regularity conditions inference for parameters $\boldsymbol{\beta}_{jk}$ can be based on the asymptotic normality of the posterior distribution

$$\boldsymbol{\beta}_{jk} \,|\, \mathbf{y} \overset{a}{\sim} \mathcal{N}\left(\hat{\boldsymbol{\beta}}_{jk}, \mathbf{H}(\hat{\boldsymbol{\beta}}_{jk})^{-1}\right),$$

with $\hat{\boldsymbol{\beta}}_{jk}$ as the posterior mode estimate. However, this approach is problematic since it does not take into account the uncertainty of estimated smoothing parameters. Moreover, from a computational perspective it can be difficult to derive the full Hessian information, because this might involve complex cross derivatives of the parameters and there are cases where standard numerical techniques cannot be applied (see Section 6.2).

Instead, applying fully Bayesian inference is relatively easy by direct computation of desired statistics from posterior samples. Computational costs are relatively low, since only samples for parameters $\boldsymbol{\beta}_{jk}$ and $\boldsymbol{\tau}^2_{jk}$ need to be saved (in practice about 2000–3000 are sufficient) from which inference of any combination of terms is straightforward, too.

The posterior predictive distribution is approximated similarly. Random samples for response observations given new covariate values $\mathbf{x}^\star$ are computed by drawing samples from the response distribution

$$y^\star \sim \mathcal{D}\left(h_1(\theta_1) = \eta_1(\mathbf{x}^\star; \boldsymbol{\beta}_1^{(t)}),\ \ldots,\ h_K(\theta_K) = \eta_K(\mathbf{x}^\star; \boldsymbol{\beta}_K^{(t)})\right)$$

for each posterior sample $\boldsymbol{\beta}_k^{(t)}$; $k = 1, \ldots, K$, $t = 1, \ldots, T$.

5. Strategies for implementation

An implementation of the conceptional framework proposed in the previous sections is provided in the R package **bamlss** (Umlauf *et al.* 2017). In this section, we outline the strategies that have been guiding this implementation but technical and R-specific details are kept brief. Instead we focus on how the flexible conceptual framework with its "Lego bricks" can be turned into an extensible and modular computational framework that readily allows to contruct estimation algorithms as well as interfaces to existing software packages such as **JAGS** (Plummer 2003) or **BayesX** (Belitz *et al.* 2017).

To provide a common toolbox that allows to play with the Lego bricks introduced in the previous sections, a general BAMLSS software system can be set up as shown in Figure 1. This proceeds in the following steps:

1. A unified model description where a `formula` specifies how to set up the linear predictors from the `data` and the `family` provides information about the Lego bricks B1–B8.

2. A generic method for setting up model terms and a `model.frame` along with the corresponding prior structures. A `transformer` can optionally set up modified terms, e.g., design and penalty matrices for smooth terms when using the mixed model representation (3).

3. Support for modular and exchangeable updating functions or complete model fitting engines in order to optionally implement either E1 or E2. First, an (optional) `optimizer` function can be run, e.g., for computing posterior mode estimates (E1) using Algorithm A1 and A2a. Second, a `sampler` is employed for full Bayesian inference with MCMC using Algorithm A1 in combination with A2b, which uses the posterior mode estimates from the `optimizer` as staring values. An additional step can be used for preparing the `results`.

4. Standard post-modeling extractor functions to create sampling statistics, visualizations, predictions, etc.

The items above are then essentially just collected in the main model fitting function called `bamlss()`. The most important arguments are

```
bamlss(formula, family = "gaussian", data = NULL,
  weights = NULL, subset = NULL, offset = NULL, na.action = na.omit,
  transform = NULL, optimizer = NULL, sampler = NULL, results = NULL,
  start = NULL, ...)
```

where the first two lines basically represent the standard model frame specifications (see Chambers and Hastie 1992).

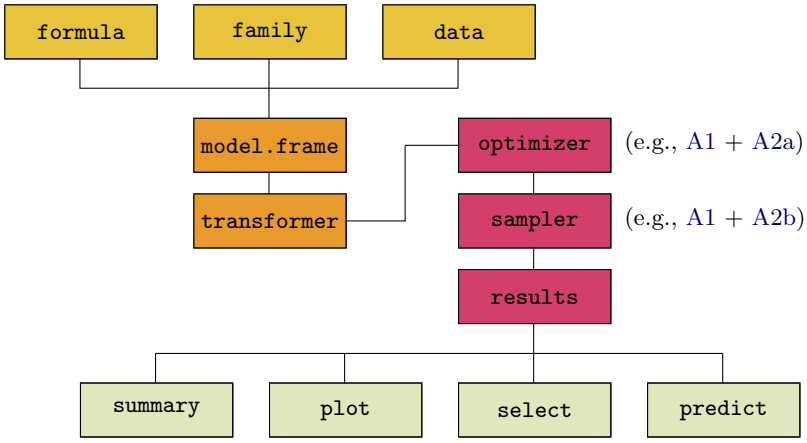

Figure 1: Flexible BAMLSS architecture.

The `formula` combines the classic Wilkinson and Rogers (1973) symbolic description – used in most standard R regression functions (Chambers and Hastie 1992) – with the infrastructure for smooth model terms like `s()`, `te()`, `ti()`, etc. – based on recommended R package **mgcv** (Wood 2016b) – and handling multiple additive predictors – utilizing the extended formula processing of Zeileis and Croissant (2010). Thus, a formula can be as simple as in a typical linear regression model with a response variable `y` and regressors `x1` and `x2`

```
y ~ x1 + x2
```

but also with smooth terms in further covariates `x3`, `x4`, and `x5`

```
y ~ x1 + x2 + s(x3) + s(x4, x5)
```

or even with different additive predictors for different model parameters, e.g.,

```
list(
   y ~ x1 + x2 + s(x3) + s(x4),
   sigma ~ x1 + x2 + s(x3)
)
```

in a normal model with $y \sim \mathcal{N}(\mu = \eta_\mu, \log(\sigma) = \eta_\sigma)$.

Similarly to other flexible model fitting functions users can specify their own `family` objects in order to plug in different Lego bricks for B1–B8. Family objects from the **gamlss** package (Stasinopoulos and Rigby 2016) are readily supported.

Estimation is performed by an `optimizer` and/or `sampler` function, which can be provided by the user. The default optimizer function implements the IWLS backfitting algorithm (16) with automatic smoothing variance selection, see also Algorithm A2a. The default sampler function implements derivative-based MCMC using IWLS proposals, smoothing variances are sampled using slice sampling, see also Section 4. For writing new optimizer and sampler functions only a simple general format of function arguments and return values must be adhered to.

More technical details are deferred to the documentation manual of package **bamlss**.

6. Illustrations

6.1. Censored heteroscedastic precepitation climatolgy from daily data

Climatology models are one important component of the meteorological tool set. The accurate and complete knowledge of precipitation climatologies is especially relevant for problems involving agriculture, risk assessment, water management, tourism etc. One particular challenge of such models is the prediction at high temporal and spatial resolutions, especially in areas without measurement. This is usually accounted for by simple averaging/smoothing at a coarse temporal scale (e.g., monthly aggregations) combined with a second step using spatial interpolation methods like Kriging (Krige 1951). However, such approaches may not work well enough at a daily resolution where precipitation data is skewed and exhibits high density at zero observations. To address these issues, Stauffer, Messner, Mayr, Umlauf, and Zeileis (2017) have recently suggested an additive regression model for daily precipitation observations based on a censored normal response distribution and various smooth spatio-temporal effects.

Following the model of Stauffer *et al.* (2017) for the province of Tyrol in Austria, we take their approach a step further and establish a daily precipitation climatology for all of Austria using a large and freely-available homogenized data source. The data are taken from the HOMSTART project (`http://www.zamg.ac.at/cms/de/forschung/klima/datensaetze/homstart/`) conducted at the Zentralanstalt für Meteorologie und Geodynamik (ZAMG) and funded by the Austrian Climate Research Programme (ACRP, Nemec, Chimani, Gruber, and Auer 2011; Nemec, Gruber, Chimani, and Auer 2013). Homogenization was successfully carried out for daily precipitation time series within 1948–2009 from a rather dense net of 57 meteorological stations (see the left panel of Figure 2). Umlauf, Mayr, Messner, and Zeileis (2012) previously investigated the data based on a much simpler ordered probit model to answer the question whether it rains more frequently on weekends than during work days (it does not). Here, we reanalyze the data using a much more complex additive regression model with a normal response left-censored at zero. To make positive observations more "normal", a commonly-used square-root transformation has been applied prior to regression modeling (see the right panel of Figure 2).

Specifically, the censored normal model with latent Gaussian variable $y^\star$ and observed response y, the square root of daily precipitation observations, is given by

$$\begin{aligned} y^\star &\sim \mathcal{N}(\mu, \sigma^2) \\ \mu &= \eta_\mu \\ \log(\sigma) &= \eta_\sigma \\ y &= \max(0, y^\star). \end{aligned}$$

Because precipitation in the Alps is driven by the season and local characteristics, e.g., differing altitude form north to south, we use the following additive predictor for parameter μ and σ:

$$\eta = \beta_0 + f_1(\texttt{alt}) + f_2(\texttt{day}) + f_3(\texttt{lon}, \texttt{lat}) + f_4(\texttt{day}, \texttt{lon}, \texttt{lat}),$$

here function f_1 is an altitude effect, f_2 the cyclic seasonal variation, f_3 a spatially correlated effect of longitude and latitude coordinates and f_4 a spatially-varying seasonal effect. Hence, the overall seasonal effect is constructed by the main effect f_2 and the interaction effect f_4,

where the deviations from the main effect are modeled to sum to zero for each day of the year, i.e., this can be viewed as a functional ANOVA decomposition.

For full Bayesian estimation with Algorithm A1, A2a and A2b, we construct updating functions $U_{jk}(\cdot)$ based on IWLS structures. Hence, as shown in Section 4 this only requires the following "Lego bricks" to be implemented:

B1. The density function of a left censored Gaussian distribution with the threshold at zero is given by

$$f(y;\, \mu = \eta_\mu,\, \log(\sigma) = \eta_\sigma) = \begin{cases} \frac{1}{\sigma}\phi\left(\frac{y-\mu}{\sigma}\right) & y > 0 \\ \Phi\left(\frac{-\mu}{\sigma}\right) & \text{else,} \end{cases} \tag{18}$$

where ϕ is the probability density and Φ the cumulative distribution function of the standard normal distribution.

B6b. Score vectors $\mathbf{u}_k = \partial\ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})/\partial\boldsymbol{\eta}_k$ are computed with

$$\frac{\partial\ell(\boldsymbol{\beta}; y, \mathbf{x})}{\partial\eta_\mu} = \begin{cases} \frac{y-\mu}{\sigma^2} & y > 0 \\ -\frac{1}{\sigma}\frac{\phi\left(\frac{-\mu}{\sigma}\right)}{\Phi\left(\frac{-\mu}{\sigma}\right)} & \text{else,} \end{cases}$$

and

$$\frac{\partial\ell(\boldsymbol{\beta}; y, \mathbf{x})}{\partial\eta_\sigma} = \begin{cases} -1 + \frac{(y-\mu)^2}{\sigma^2} & y > 0 \\ -\frac{-\mu}{\sigma}\frac{\phi\left(\frac{-\mu}{\sigma}\right)}{\Phi\left(\frac{-\mu}{\sigma}\right)} & \text{else.} \end{cases}$$

B7b. The diagonal elements of the weight matrix $\mathbf{W}_{kk} = -\text{diag}(\partial^2\ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})/\partial\boldsymbol{\eta}_k\partial\boldsymbol{\eta}_k^\top)$ are derived using

$$\frac{\partial^2\ell(\boldsymbol{\beta}; y, \mathbf{x})}{\partial\eta_\mu^2} = \begin{cases} -\frac{1}{\sigma^2} & y > 0 \\ -\frac{-\mu}{\sigma^3}\frac{\phi\left(\frac{-\mu}{\sigma}\right)}{\Phi\left(\frac{-\mu}{\sigma}\right)} - \frac{1}{\sigma^2}\frac{\phi\left(\frac{-\mu}{\sigma}\right)^2}{\Phi\left(\frac{-\mu}{\sigma}\right)^2} & \text{else,} \end{cases}$$

and

$$\frac{\partial^2\ell(\boldsymbol{\beta}; y, \mathbf{x})}{\partial\eta_\sigma^2} = \begin{cases} -2\frac{(y-\mu)^2}{\sigma^2} & y > 0 \\ -\frac{-\mu}{\sigma}\frac{\phi\left(\frac{-\mu}{\sigma}\right)}{\Phi\left(\frac{-\mu}{\sigma}\right)} - \frac{(-\mu)^3}{\sigma^3}\frac{\phi\left(\frac{-\mu}{\sigma}\right)}{\Phi\left(\frac{-\mu}{\sigma}\right)} - \frac{(-\mu)^2}{\sigma^2}\frac{\phi\left(\frac{-\mu}{\sigma}\right)^2}{\Phi\left(\frac{-\mu}{\sigma}\right)^2} & \text{else.} \end{cases}$$

The first and second derivative functions have been implemented in the **bamlss** family `cnorm_bamlss()`.

Since the HOMSTART data set has over 1.2 million observations, the full storage of the resulting design matrices would lead to excessive demands concerning both computer storage as well as CPU power. In order to prevent computational problems associated with very large data sets like HOMSTART, we make use of the fact that the number of unique regressor observations is much smaller, e.g., only 365 for the day-of-year effect. This is much smaller than the total number of observations of the data set and duplicated rows in the corresponding design matrix can be avoided within the model fitting algorithms. Therefore, we implemented updating functions $U_{jk}(\cdot)$ that support shrinkage of the design matrices based on unique covariate observations, using the highly-efficient algorithm of Lang *et al.* (2014). This essentially employs a reduced form of the diagonal weight matrix $\mathbf{W}_{kk}$ in the IWLS algorithm and computes the reduced partial residual vector from $\mathbf{z}_k - \boldsymbol{\eta}_{k,-j}^{(t)}$ separately. For

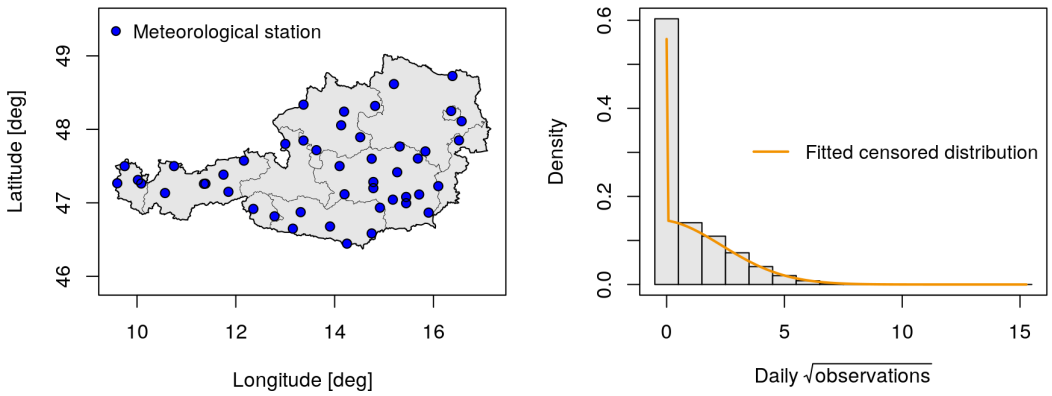

Figure 2: Distribution of available meteorological stations and daily precipitation values.

usage within **bamlss** see also the documentation of estimation engines `bfit()` and `GMCMC()` and the corresponding updating functions `bfit_iwls()` and `GMCMC_iwls()`.

With a total of 4000 iterations of the MCMC sampler, on a Linux system with 8 Intel i7-2600 3.40GHz processors running the model takes approximately 17 hours. For computing the final model output the first 2000 samples of every core are withdrawn and only every 10th sample is saved.

The plots of the estimated effects are shown in Figure 3. The top row illustrates the spatial variation of the seasonal effect (solid lines) together with the mean effect (dashed lines) for parameters μ and σ. The estimates indicate that during June to August precipitation is highest in the mean effect for μ. However, there is some clear spatial variation, especially differences between the regions north and south of the Alps. This is highlighted by the red, gray and blue lines and show that the southern stations have a clear annual peak while for the northern stations the semiannual pattern is more pronounced. Similarly, the seasonal effect for parameter σ has considerable variation between north and south. The uncertainty peak is shifting from the middle of summer to fall when going from north to south.

The second row of Figure 3 shows the resulting spatial trends. The spatial effect for parameter μ indicates that regions with positive effect accumulate in the north-west part of Austria. The overall importance of the spatial effect is somewhat smaller compared to the seasonal effects, which is highlighted by using the same range for y-axes in the first row and the color legends in the second row. The spatial effect for parameter σ shows that model uncertainty is the highest within the southern regions (especially the province of Carinthia) and in the most western province (Vorarlberg).

The bottom plot in Figure 3 is an example of the resulting precipitation climatology for January 10th. The predicted average precipitation is quite low all over Austria, ranging from 0 to 1.1mm. The map indicates that more precipitation can be expected in the northern parts of the Alps, especially in the west (Vorarlberg) and in the center (Salzburg). The effect of elevation is also visible since the valleys exhibit less precipitation than the alpine regions, however, the effect is not as pronounced as, e.g., the seasonal effect(s), most probably because the variation of elevation of the meteorological stations used in this data set is relatively small.

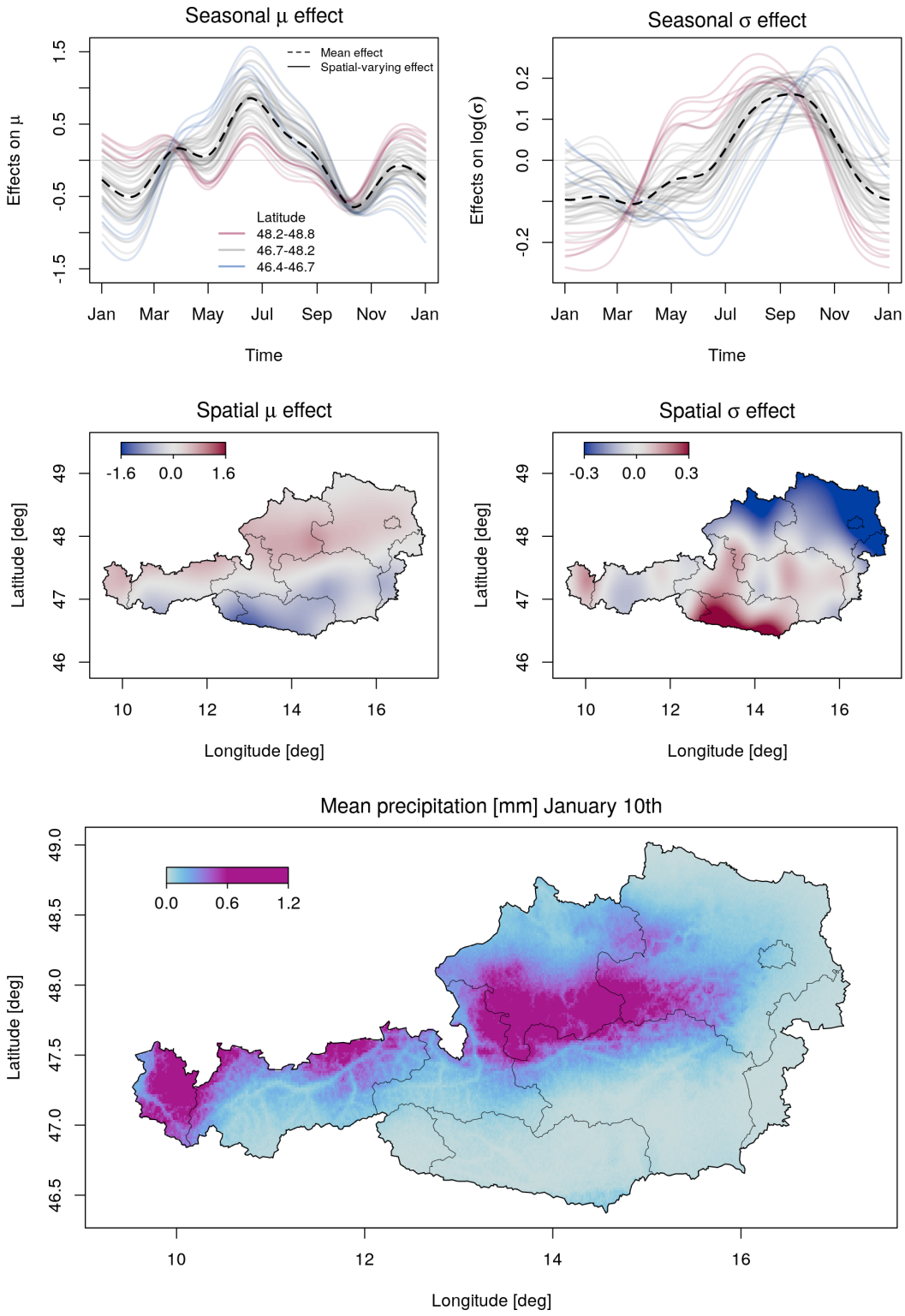

Figure 3: Estimated effects of the precipitation model, 1st and 2nd row, predicted average precipitation of the censored mean computed using sampling from the fitted distribution for January 10th, bottom row.

6.2. Complex space-time interactions in a Cox model

This analysis is based on the article of Taylor (2017) and contributes to the developed model by inclusion of complex space-time interactions using the BAMLSS framework.

The *London Fire Brigade* (LFB, http://www.london-fire.gov.uk/) is one of the largest in the world. Each year, the LFB is called thousands of times, in most cases due to dwelling fires. To prevent further damage or fatal casualties, a short arrival time is important, i.e., the time it takes until a fire engine arrives at the scene after an emergency call has been received. The LFB's annual performance target is an average fire engine arrival time of six minutes at maximum. Clearly, this mostly depends on the distance between the site and the responsible fire station but it may also depend on the number of fire stations in the area because fire engines may already be in use at another nearby fire scenery. Therefore, Taylor (2017) analyzes the effect of fire station closures in 2014 using a parametric proportional hazards model to identify regions of possible concern about the number of available fire stations. To contribute to the topic, we apply an extended complex Cox model to the 2015 dwelling fire response time data and illustrate how the generic BAMLSS framework can be utilized to set up new estimation algorithms for this type of model.

The data is freely available from the London DataStore (http://data.london.gov.uk/) under the UK Open Government Licence (OGL v2). It can be downloaded from http://data.london.gov.uk/dataset/london-fire-brigade-incident-records which also contains previous year.

The dwelling fire data for 2015 consists of 5838 fire events that have been recorded at the 103 fire stations. The distribution of dwelling fires and fire stations is shown in Figure 4. The top left panel indicates that both, fire stations and fire events, are spread all over London with a higher density in the city center which is brought out more clearly by the heatmap in the bottom left panel. The panels on the right-hand side pertain to the arrival time and show that overall about 30% of these were greater than six minutes (bottom right) with most of these occuring at the borders of London (top right).

Taylor (2017) analyzes the response times within a survival context where the hazard of an event (fire engine arriving) at time t with a relative risk model of the form

$$\lambda(t) = \exp\left(\eta(t)\right) = \exp\left(\eta_\lambda(t) + \eta_\gamma\right),$$

i.e., a model for the instantaneous arrival rate conditional on the engine not having arrived before time t. Here, the hazard function is assumed to depend on a time-varying predictor $\eta_\lambda(t)$ and a time-constant predictor η_γ. In most survival models, the time-varying part $\eta_\lambda(t)$ represents the so-called baseline hazard and is a univariate function of time t. Compared to Taylor (2017), we set up a similar model but with the extended time-constant predictor

$$\eta_\gamma = \beta_0 + f_1(\texttt{fsintens}) + f_2(\texttt{daytime}) + f_3(\texttt{lon}, \texttt{lat}) + f_4(\texttt{daytime}, \texttt{lon}, \texttt{lat}),$$

where β_0 is an intercept and function f_1 is the effect of fire station intensity (`fsintens`, computed with a kernel density estimate of all fire stations in London). Thus, this variable is a proxy for the distance to the next fire station(s), especially suited for situations when the responsible fire station already send out all fire engines such that help needs to arrive from another station. Function f_2 accounts for the effect that it is more difficult for a fire engine to arrive at the scene in rush hours, i.e., the risk of waiting longer than six minutes is expected

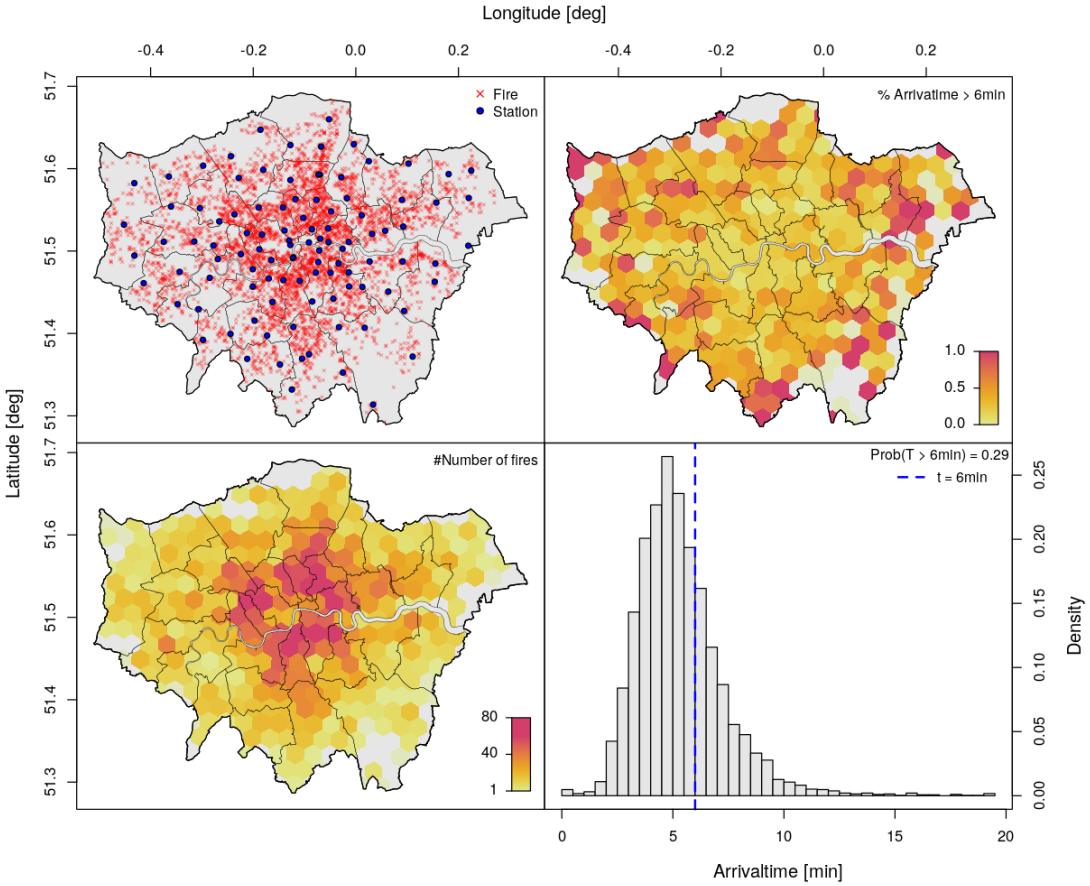

Figure 4: Distribution of dwelling fires, fire stations, and arrival times in London, 2015.

to depend on the time of the day, variable `daytime`. To treat the question of structured spatially driven hazards, a spatial effect f_3 of longitude and latitude coordinates is included in the model. Moreover, we also treat the `daytime` effect in a spatially correlated context, function f_4. For example, we assume that rush hour peaks may have local hot spots that can be captured by this three-dimensional effect. Again, all functions $f_1, \ldots, f_4$ are assumed to be possibly nonlinear and are modeled using penalized splines.

Moreover, we also relax the time-varying predictor $\eta_\lambda(t)$ to

$$\eta_\lambda(t) = f_0(t) + \sum_{j=1}^{J_\lambda} f_j(t, \mathbf{x}).$$

Here, the baseline hazard is represented by $f_0(t)$ and all functions $f_j(t, \mathbf{x})$ are time-varying possibly nonlinear functions of covariates. Hence, our model is a complex Cox-type additive model as introduced by Kneib and Fahrmeir (2007). To further investigate if there is a space-time varying effect, i.e., if the shape of the baseline hazard is dependent on the location we use the following time-varying additive predictor

$$\eta_\lambda(\texttt{arrivaltime}) = f_0(\texttt{arrivaltime}) + f_1(\texttt{arrivaltime}, \texttt{lon}, \texttt{lat}),$$

where f_0 is the baseline hazard for variable `arrivaltime`, the waiting time until the first

fire engine arrives after the received emergency call. Function $f_1(\texttt{arrivaltime}, \texttt{lon}, \texttt{lat})$ is a space-time varying effect modeling the deviations from the baseline which can capture whether the risk of waiting longer than six minutes is driven by other factors that are not available in this analysis. Both functions are modeled using penalized splines.

The probability that the engine will arrive on the scene after time t is described by the survival function

$$S(t) = \text{Prob}(T > t) = \exp\left(-\int_0^t \lambda(u)du\right), \tag{19}$$

which is of prime interest in this analysis. Based on (19), for full Bayesian inference the following "Lego bricks" need to be implemented for updating functions $U_{jk}(\cdot)$ using algorithms A1, A2a and A2b:

B1. The log-likelihood function of the continuous time Cox model is given by

$$\ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X}) = \sum_{i=1}^{n}\left(\delta_i \eta_{i\gamma} - \int_0^{t_i} \exp(\eta_{i\lambda}(u)du)\right).$$

where δ_i is the usual censoring indicator, which equals to $\delta_i = 1$ in this example, because we focus on real fire events.

B6a. For derivative-based estimation using Algorithm A2a and for MCMC simulation with Algorithm A2b, the score vectors and Hessian need to be computed. Assuming a basis function approach, the score vector of the regression coefficients for the time-varying part $\eta_\lambda(t)$ is

$$\mathbf{s}\left(\boldsymbol{\beta}_\lambda\right) = \boldsymbol{\delta}^\top \mathbf{X}_\lambda(\mathbf{t}) - \sum_{i=1}^{n} \exp(\eta_{i\gamma})\left(\int_0^{t_i} \exp(\eta_{i\lambda}(u))\mathbf{x}_i(u)du\right).$$

B7a. The elements of the Hessian w.r.t. $\boldsymbol{\beta}_\lambda$ are

$$\mathbf{H}\left(\boldsymbol{\beta}_\lambda\right) = -\sum_{i=1}^{n} \exp\left(\eta_{i\gamma}\right) \int_0^{t_i} \exp(\eta_{i\lambda}(u))\boldsymbol{x}_{i\lambda}(u)\boldsymbol{x}_{i\lambda}^\top(u)du.$$

Note that the Hessian cannot be fragmented further to obtain building block B7b and IWLS updating functions. The reason is that the diagonal weight matrix based on $\partial^2\ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})/\partial\boldsymbol{\eta}_\lambda(\mathbf{t})\partial\boldsymbol{\eta}_\lambda(\mathbf{t})^\top$ requires a functional derivative like the Hadamard derivative since the predictor depends on time t. However, it turns out that this derivative forms martingale residuals in the IWLS step (see, e.g., Barlow 1988) which are incapable of estimating time-varying effects, see also Hofner (2008, Section 5.2) for a detailed discussion. Therefore updating functions $U_{jk}(\cdot)$ for the time-varying predictor $\eta_\lambda(t)$ are based on updating Equation (17) within Algorithm A2a and A2b.

B6b & B7b. Constructing updating functions for the time-constant part η_γ again yields an IWLS updating scheme, see Section 4, with working observations given by

$$\mathbf{z} = \boldsymbol{\eta}_\gamma + \mathbf{W}^{-1}\mathbf{u},$$

with the weight matrix

$$\mathbf{W} = \text{diag}(\mathbf{P}\exp(\boldsymbol{\eta}_\gamma)),$$

where $\mathbf{P}$ is a diagonal matrix with elements $p_{ii} = \int_0^{t_i} \exp(\eta_{i\lambda}(u)du)$. The score vector is

$$\mathbf{u} = \boldsymbol{\delta} - \mathbf{P}\exp(\boldsymbol{\eta}_\gamma).$$

(Hennerfeind, Brezger, and Fahrmeir 2006)

As a result, applying the generic algorithm presented in Algorithm A1 to this type of problem two specific difficulties need to be considered. First, the updating functions $U_{jk}(\cdot)$ for the time-varying predictor $\eta_\lambda(t)$ are different from the time-constant updating functions for η_γ. Secondly, a specific hurdle of the continuous-time Cox model is the computation of the integrals, because these do not have a closed form solution and need to be approximated numerically, e.g., by the trapezoidal rule or Gaussian quadrature (Hofner 2008; Waldmann, Taylor-Robinson, Klein, Kneib, Pressler, Schmid, and Mayr 2016). Moreover, it is inefficient to compute the integrals anew for every updating step, since for the time-constant part the integrals given in $\mathbf{P}$ do not change anymore.

In order to reduce computing time we account for the idiosyncrasy of the Cox model and implement an optimizer function `cox.mode()` for posterior mode estimation as well as the sampler function `cox.mcmc()` for MCMC simulation. The amount of work to implement this model using the **bamlss** infrastructures is moderate, because most of the code of the default estimation engines can be reused and only need slight adaption. In this example, the the optimizer and sampler function are part of the corresponding **bamlss** family object `cox_bamlss()`. On a Linux system with 8 Intel i7-2600 3.40GHz processors estimation takes approximately 1.2 days. Note that function `cox.mode()` also applies an automated procedure for smoothing variances selection using information criteria, see also Algorithm A2a.

The estimated effects are shown in Figure 5. The upper left panel shows that the average "risk" that a fire engine arrives increases steeply until the target time of six minutes. The space-time varying effect is relatively small compared to the overall size of the effect, especially until the six minutes target time it seems that the location does not have a great influence on the relative risk. Only for waiting times above ∼15 minutes, the space-time varying effect is more pronounced. The effect for fire station intensity is quite large and bounded, i.e., there is a natural limit for the benefit from opening new fire stations in the area. The effect of the time of the day then indicates that in the morning hours around 4–5 am, as well as in the afternoon around 2–4 pm, the risk of waiting longer for the fire engine to arrive is only slightly increasing. In addition, the spatial deviations from the mean time of day effect are modest, similar in magnitude as the spatial varying baseline effects. The largest deviation seems to be at around 10 am. In Figure 6 the spatial varying effect is illustrated on 9 time points. The maps indicate possible hot-spots of this effect, however, as mentioned above the overall effect size from -0.4 to 0.4 is not very large (see also Figure 5 bottom left) such that differences in risk probabilities are almost negligible. In contrast, the time-constant spatial effect clearly shows that the average risk of increased waiting times are higher in the city center and some smaller area in southern London. However, the estimated probabilities of waiting longer than six minutes around the center show moderate variation, while the borders of London indicate higher probabilities as well as in the western parts, most probably because of the lower fire station density in these areas. In summary, next to the baseline effect, the most important effects on the log risk are the fire station intensity and the time-constant spatial effect which have an absolute range of about 4 on the log-scale.

To conclude, the proposed model including complex model terms beyond "classical" structures, like space-time interactions in both the time-constant and the time-varying part, is a

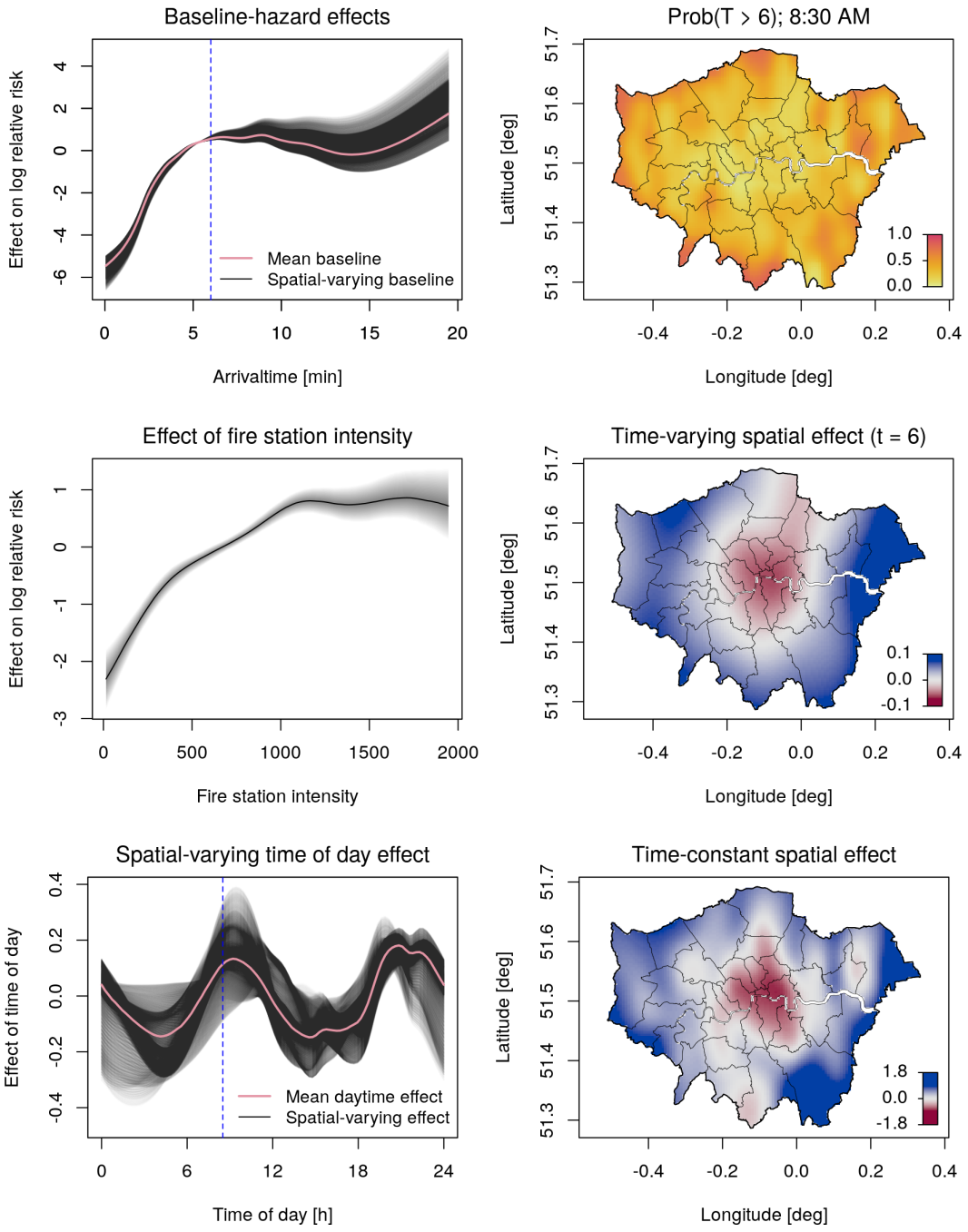

Figure 5: Estimated effects of the fire emergency response times survival model. Top left panel shows the mean baseline effect, red line, together with the spatially-varying effects, black lines. The six minutes target waiting time is represented by the blue dashed vertical line. The upper right panel shows the estimated probability of waiting longer than six minutes until the first engine arrives at 8:30 am. The space-time varying effect is illustrated at six minutes waiting time in the second row, right panel. The time of day effect again shows the mean effect as red lines and spatial deviations by black lines.

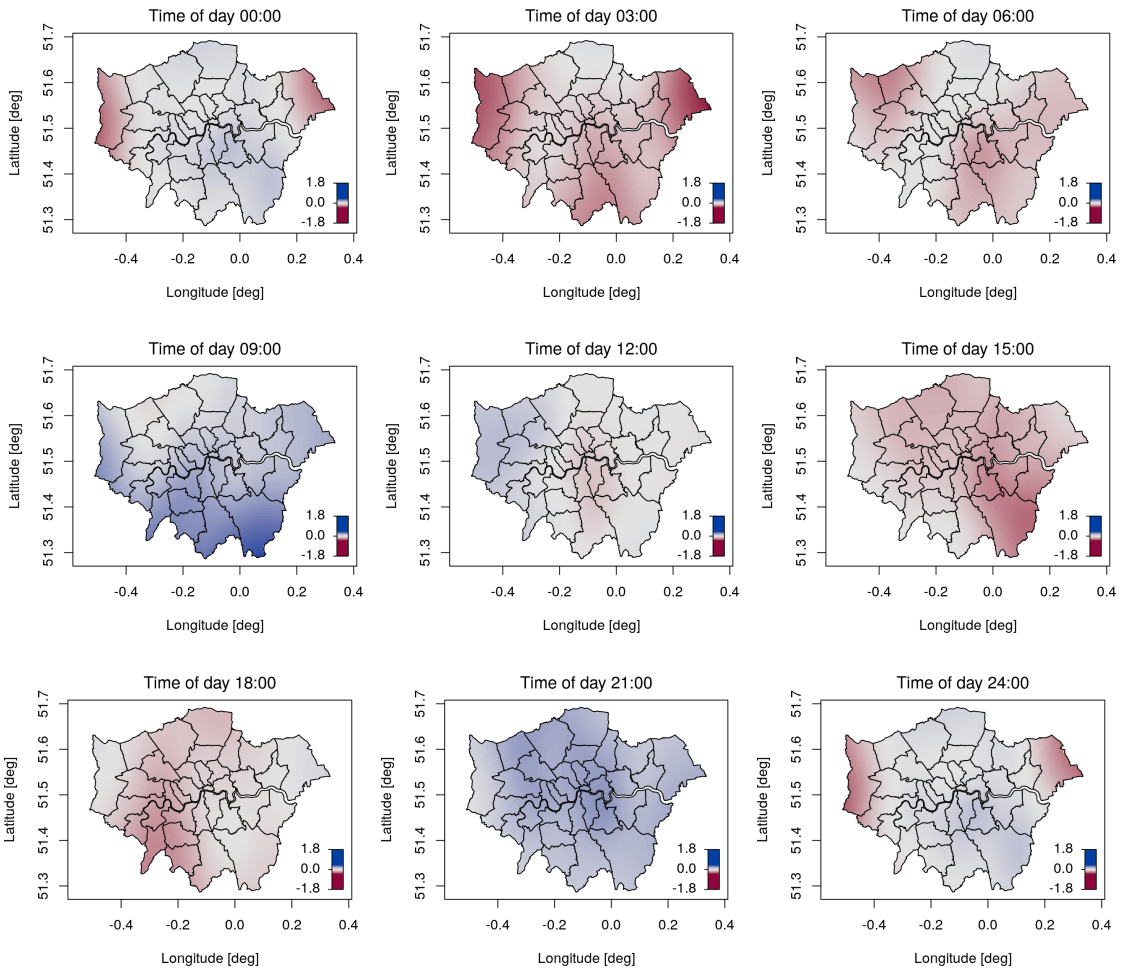

Figure 6: Estimated spatial varying time-of-the-day effect.

considerable extension of this type of model and can gain more insight into potential risk factors that are probably not obvious. The presented modular framework facilitates the development of such complex algorithms essentially, e.g., Köhler, Umlauf, Beyerlein, Winkler, Ziegler, and Greven (2016) develop flexible joint models for longitudinal and time-to-event data using the modular BAMLSS framework.

7. Summary

This paper combines frequently-used algorithms for the estimation of additive Bayesian models in a flexible framework for distributional regression, also termed Bayesian additive models for location, scale and shape (BAMLSS), and beyond. We highlight the similarities between optimization and sampling concepts and coalesce these in a generic toolbox of modular "Lego bricks". Two case studies illustrate how the framework can be leveraged to establish complex and difficult-to-estimate models based on the accompanying implementation in the R package **bamlss** (Umlauf *et al.* 2017).

References

Barlow RLPWE (1988). "Residuals for Relative Risk Regression." *Biometrika*, **75**(1), 65–74.

Belitz C, Brezger A, Kneib T, Lang S, Umlauf N (2017). ***BayesX*** *– Software for Bayesian Inference in Structured Additive Regression Models*. Version 3.0.2, URL http://www.BayesX.org/.

Belitz C, Lang S (2008). "Simultaneous Selection of Variables and Smoothing Parameters in Structured Additive Regression Models." *Computational Statistics & Data Analysis*, **53**, 61–81. doi:10.1016/j.csda.2008.05.032.

Brezger A, Lang S (2006). "Generalized Structured Additive Regression Based on Bayesian P-Splines." *Computational Statistics & Data Analysis*, **50**, 947–991. doi:10.1016/j.csda.2004.10.011.

Carpenter B, Gelman A, Hoffman MD, Lee D, Goodrich B, Betancourt M, Brubaker M, Guo J, Li P, Riddell A (2017). "Stan: A Probabilistic Programming Language." *Journal of Statistical Software*, **76**(1), 1–32. doi:10.18637/jss.v076.i01.

Chambers JM, Hastie TJ (eds.) (1992). *Statistical Models in S*. Chapman & Hall, London.

Dunn PK, Smyth GK (1996). "Randomized Quantile Residuals." *Journal of Computational and Graphical Statistics*, **5**, 236–245. doi:10.2307/1390802.

Fahrmeir L, Kneib T, Lang S (2004). "Penalized Structured Additive Regression for Space Time Data: A Bayesian Perspective." *Statistica Sinica*, **14**, 731–761. doi:10.1007/978-3-642-34333-9_9.

Fahrmeir L, Kneib T, Lang S, Marx B (2013). *Regression – Models, Methods and Applications*. Springer-Verlag, Berlin. ISBN 978-3-642-34332-2.

Gamerman D (1997). "Sampling from the Posterior Distribution in Generalized Linear Mixed Models." *Statistics and Computing*, **7**(1), 57–68. ISSN 0960-3174. doi:10.1023/a:1018509429360.

Gelman A (2006). "Prior Distributions for Variance Parameters in Hierarchical Models (Comment on Article by Browne and Draper)." *Bayesian Analysis*, **1**(3), 515–534. doi:10.1214/06-ba117a.

Gelman A, Roberts GO, Gilks WR (1996). "Efficient Metropolis Jumping Rules." In JM Bernado, others (eds.), *Bayesian Statistics*, volume 5, p. 599. OUP.

Gneiting T, Balabdaoui F, Raftery AE (2007). "Probabilistic Forecasts, Calibration and Sharpness." *Journal of the Royal Statistical Society B*, **69**(2), 243–268. ISSN 1467-9868. doi:10.1111/j.1467-9868.2007.00587.x.

Hastie T, Tibshirani R (1990). *Generalized Additive Models*. Chapman & Hall/CRC.

Hennerfeind A, Brezger A, Fahrmeir L (2006). "Geoadditive Survival Models." *Journal of the American Statistical Association*, **101**(475), 1065–1075. doi:10.1198/016214506000000348.

Hofner B (2008). *Variable Selection and Model Choice in Survival Models with Time-Varying Effects*. Ph.D. thesis, Institut für Statistik. URL http://nbn-resolving.de/urn/resolver.pl?urn=nbn:de:bvb:19-epub-11028-5.

Klein N, Kneib T (2016a). "Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression." *Bayesian Analysis*, **11**(4), 1071–1106. doi:10.1214/15-ba983.

Klein N, Kneib T (2016b). "Simultaneous Inference in Structured Additive Conditional Copula Regression Models: A Unifying Bayesian Approach." *Statistics and Computing*, **26**(4), 841–860. ISSN 1573-1375. doi:10.1007/s11222-015-9573-6.

Klein N, Kneib T, Klasen S, Lang S (2015a). "Bayesian Structured Additive Distributional Regression for Multivariate Responses." *Journal of the Royal Statistical Society C*, **64**, 569–591. ISSN 1467-9876. doi:10.1111/rssc.12090.

Klein N, Kneib T, Lang S (2015b). "Bayesian Generalized Additive Models for Location, Scale and Shape for Zero-Inflated and Overdispersed Count Data." *Journal of the American Statistical Association*, **110**(509), 405–419. doi:10.1080/01621459.2014.912955.

Klein N, Kneib T, Lang S, Sohn A (2015c). "Bayesian Structured Additive Distributional Regression with an Application to Regional Income Inequality in Germany." *Annals of Applied Statistics*, **9**, 1024–1052. doi:10.1214/15-aoas823.

Kneib T, Fahrmeir L (2007). "A Mixed Model Approach for Geoadditive Hazard Regression." *Scandinavian Journal of Statistics*, **34**(1), 207–228. doi:10.1111/j.1467-9469.2006.00524.x.

Köhler M, Umlauf N, Beyerlein A, Winkler C, Ziegler AG, Greven S (2016). "Flexible Bayesian Additive Joint Models with an Application to Type 1 Diabetes Research." arXiv:1611.01485.

Krige DG (1951). "A Statistical Approach to Some Basic Mine Valuation Problems on the Witwatersrand." *Journal of the Chemical, Metallurgical and Mining Society of South Africa*, **52**(6), 119–139. doi:10.2307/3006914.

Lang S, Umlauf N, Wechselberger P, Harttgen K, Kneib T (2014). "Multilevel Structured Additive Regression." *Statistics and Computing*, **24**(2), 223–238. ISSN 0960-3174. doi:10.1007/s11222-012-9366-0.

Lunn DJ, Spiegelhalter D, Thomas A, Best N (2009). "The **BUGS** Project: Evolution, Critique and Future Directions." *Statistics in Medicine*, **28**, 3049–3082. doi:10.1002/sim.3680.

Lunn DJ, Thomas A, Best N, Spiegelhalter D (2000). "**WinBUGS** – A Bayesian Modelling Framework: Concepts, Structure, and Extensibility." *Statistics and Computing*, **10**, 325–337. doi:10.1023/a:1008929526011.

Neal RM (2003). "Slice Sampling." *The Annals of Statistics*, **31**(3), 705–767.

Nemec J, Chimani B, Gruber C, Auer I (2011). "Ein Neuer Datensatz Homogenisierter Tagesdaten." *ÖGM Bulletin*, (2011/1), 19–20. doi:10.1002/joc.3532.

Nemec J, Gruber C, Chimani B, Auer I (2013). "Trends in Extreme Temperature Indices in Austria Based on a New Homogenised Dataset." *International Journal of Climatology*, **33**(6), 1538–1550. doi:10.1002/joc.3532.

Plummer M (2003). "**JAGS**: A Program for Analysis of Bayesian Graphical Models Using Gibbs Sampling." In K Hornik, F Leisch, A Zeileis (eds.), *Proceedings of the 3rd International Workshop on Distributed Statistical Computing, Vienna, Austria*. ISSN 1609-395X. URL http://www.ci.tuwien.ac.at/Conferences/DSC-2003/Proceedings/.

Polson NG, Scott JG (2012). "On the Half-Cauchy Prior for a Global Scale Parameter." *Bayesian Analysis*, **7**(4), 887–902. doi:10.1214/12-ba730.

R Core Team (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL https://www.R-project.org/.

Rigby RA, Stasinopoulos DM (2005). "Generalized Additive Models for Location, Scale and Shape." *Journal of the Royal Statistical Society C*, **54**(3), 507–554. doi:10.1111/j.1467-9876.2005.00510.x.

Roberts GO, Rosenthal JS (2009). "Examples of Adaptive MCMC." *Journal of Computational and Graphical Statistics*, **18**(2), 349–367. doi:10.1198/jcgs.2009.06134.

Ruppert D, Wand MP, Carrol RJ (2003). *Semiparametric Regression*. Cambridge University Press, New York.

Simpson D, Rue H, Martins TG, Riebler A, Sørbye SH (2017). "Penalising Model Component Complexity: A Principled, Practical Approach to Constructing Priors." *Statistical Science*. Forthcoming.

Smyth G (1996). "Partitioned Algorithms for Maximum Likelihood and Other Non-Linear Estimation." *Statistics and Computing*, **6**(3), 201–216. ISSN 0960-3174. doi:10.1007/bf00140865.

Stasinopoulos M, Rigby B (2016). ***gamlss**: Generalised Additive Models for Location Scale and Shape*. R package version 5.0-1, URL https://CRAN.R-project.org/package=gamlss.

Stauffer R, Messner JW, Mayr GJ, Umlauf N, Zeileis A (2017). "Spatio-Temporal Precipitation Climatology over Complex Terrain Using a Censored Additive Regression Model." *International Journal of Climatology*. doi:10.1002/joc.4913. Forthcoming.

Taylor BM (2017). "Spatial Modelling of Emergency Service Response Times." *Journal of the Royal Statistical Society A*. doi:10.1111/rssa.12192. Forthcoming.

Umlauf N, Adler D, Kneib T, Lang S, Zeileis A (2015). "Structured Additive Regression Models: An R Interface to **BayesX**." *Journal of Statistical Software*, **63**(1), 1–46. ISSN 1548-7660. doi:10.18637/jss.v063.i21.

Umlauf N, Klein N, Zeileis A, Köhler M (2017). ***bamlss**: Bayesian Additive Models for Location Scale and Shape (and Beyond)*. R package version 0.1-1, URL http://CRAN.R-project.org/package=bamlss.

Umlauf N, Mayr G, Messner J, Zeileis A (2012). "Why Does It Always Rain on Me? A Spatio-Temporal Analysis of Precipitation in Austria." *Austrian Journal of Statistics*, **41**(1), 81–92. doi:10.1002/joc.4913.

Waldmann E, Taylor-Robinson D, Klein N, Kneib T, Pressler T, Schmid M, Mayr A (2016). "Boosting Joint Models for Longitudinal and Time-to-Event Data." To appear in Biometrical Journal, arXiv:1609.02686.

Wand MP (2003). "Smoothing and Mixed Models." *Computational Statistics*, **18**, 223–249. doi:10.1007/s001800300142.

Wilkinson GN, Rogers CE (1973). "Symbolic Description of Factorial Models for Analysis of Variance." *Journal of the Royal Statistical Society C*, **22**(3), pp. 392–399. ISSN 00359254. doi:10.2307/2346786.

Wood SN (2004). "Stable and Efficient Multiple Smoothing Parameter Estimation for Generalized Additive Models." *Journal of the American Statistical Association*, **99**, 673–686. doi:10.1198/016214504000000980.

Wood SN (2006). *Generalized Additive Models: An Introduction with R*. Chapman & Hall/CRC, Boca Raton.

Wood SN (2016a). "Just Another Gibbs Additive Modeler: Interfacing **JAGS** and **mgcv**." *Journal of Statistical Software*, **75**(7), 1–15. doi:10.18637/jss.v075.i07.

Wood SN (2016b). ***mgcv**: GAMs with GCV/AIC/REML Smoothness Estimation and GAMMs by PQL*. R package version 1.8.16, URL https://CRAN.R-project.org/package=mgcv.

Yee TW (2015). *Vector Generalized Linear and Additive Models: With an Implementation in R*. Springer-Verlag, New York.

Zeileis A, Croissant Y (2010). "Extended Model Formulas in R: Multiple Parts and Multiple Responses." *Journal of Statistical Software*, **34**(1), 1–13. doi:10.18637/jss.v034.i01.

A. Posterior mode updating based on IWLS

The following shows the steps needed to derive the iterative updating scheme based on IWLS in Section 4.2. Focusing on the j-th row of (14) gives

$$\begin{aligned}(\mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}))\boldsymbol{\beta}_{jk}^{(t+1)} + \ldots + \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{J_k k} \boldsymbol{\beta}_{J_k k}^{(t+1)} - \\ (\mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}))\boldsymbol{\beta}_{jk}^{(t)} - \ldots - \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{J_k k} \boldsymbol{\beta}_{J_k k}^{(t)} &= \mathbf{X}_{jk}^\top \mathbf{u}_k^{(t)} - \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk})\boldsymbol{\beta}_{jk}^{(t)}\end{aligned}$$

$$\begin{aligned}\mathbf{G}_{jk}(\boldsymbol{\tau}_{jk})(\boldsymbol{\beta}_{jk}^{(t+1)} - \boldsymbol{\beta}_{jk}^{(t)}) + \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} \boldsymbol{\beta}_{jk}^{(t+1)} + \ldots + \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{J_k k} \boldsymbol{\beta}_{J_k k}^{(t+1)} - \\ \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} \boldsymbol{\beta}_{jk}^{(t)} - \ldots - \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{J_k k} \boldsymbol{\beta}_{J_k k}^{(t)} &= \mathbf{X}_{jk}^\top \mathbf{u}_k^{(t)} - \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk})\boldsymbol{\beta}_{jk}^{(t)}\end{aligned}$$

$$\begin{aligned}\mathbf{G}_{jk}(\boldsymbol{\tau}_{jk})\boldsymbol{\beta}_{jk}^{(t)} + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk})(\boldsymbol{\beta}_{jk}^{(t+1)} - \boldsymbol{\beta}_{jk}^{(t)}) + \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} \boldsymbol{\beta}_{jk}^{(t+1)} + \ldots + \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{J_k k} \boldsymbol{\beta}_{J_k k}^{(t+1)} - \\ \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} \boldsymbol{\beta}_{jk}^{(t)} - \ldots - \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{J_k k} \boldsymbol{\beta}_{J_k k}^{(t)} &= \mathbf{X}_{jk}^\top \mathbf{u}_k^{(t)}\end{aligned}$$

$$\begin{aligned}\mathbf{G}_{jk}(\boldsymbol{\tau}_{jk})\boldsymbol{\beta}_{jk}^{(t+1)} + \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} \boldsymbol{\beta}_{jk}^{(t+1)} + \ldots + \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{J_k k} \boldsymbol{\beta}_{J_k k}^{(t+1)} - \\ \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} \boldsymbol{\beta}_{jk}^{(t)} - \ldots - \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{J_k k} \boldsymbol{\beta}_{J_k k}^{(t)} &= \mathbf{X}_{jk}^\top \mathbf{u}_k^{(t)}\end{aligned}$$

$$\mathbf{G}_{jk}(\boldsymbol{\tau}_{jk})\boldsymbol{\beta}_{jk}^{(t+1)} + \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} \boldsymbol{\beta}_{jk}^{(t+1)} + \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \boldsymbol{\eta}_{k,-j}^{(t+1)} - \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \boldsymbol{\eta}_k^{(t)} = \mathbf{X}_{jk}^\top \mathbf{u}_k^{(t)}$$

$$(\mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}))\boldsymbol{\beta}_{jk}^{(t+1)} = \mathbf{X}_{jk}^\top \mathbf{u}_k^{(t)} + \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \boldsymbol{\eta}_k^{(t)} - \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \boldsymbol{\eta}_{k,-j}^{(t+1)}$$

$$\boldsymbol{\beta}_{jk}^{(t+1)} = (\mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}))^{-1}(\mathbf{X}_{jk}^\top \mathbf{u}_k^{(t)} + \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \boldsymbol{\eta}_k^{(t)} - \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \boldsymbol{\eta}_{k,-j}^{(t+1)})$$

$$\boldsymbol{\beta}_{jk}^{(t+1)} = (\mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}))^{-1} \mathbf{X}_{jk}^\top (\mathbf{u}_k^{(t)} + \mathbf{W}_{kk} \boldsymbol{\eta}_k^{(t)} - \mathbf{W}_{kk} \boldsymbol{\eta}_{k,-j}^{(t+1)})$$

$$\boldsymbol{\beta}_{jk}^{(t+1)} = (\mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}))^{-1} \mathbf{X}_{jk}^\top (\mathbf{W}_{kk} \mathbf{W}_{kk}^{-1} \mathbf{u}_k^{(t)} + \mathbf{W}_{kk} \boldsymbol{\eta}_k^{(t)} - \mathbf{W}_{kk} \boldsymbol{\eta}_{k,-j}^{(t+1)})$$

$$\boldsymbol{\beta}_{jk}^{(t+1)} = (\mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}))^{-1} \mathbf{X}_{jk}^\top \mathbf{W}_{kk} (\mathbf{W}_{kk}^{-1} \mathbf{u}_k^{(t)} + \boldsymbol{\eta}_k^{(t)} - \boldsymbol{\eta}_{k,-j}^{(t+1)})$$

This yields the updating function $U_{jk}(\cdot)$ shown in (16).

B. Approximate full conditionals for derivative-based MCMC

The following shows the steps to derive a multivariate normal jumping distribution based on a second order Taylor series expansion of the log-posterior centered at the last state of $\boldsymbol{\beta}_{jk}$.

$$\begin{aligned}
\pi(\boldsymbol{\beta}_{jk}^{\star}|\cdot) &\propto \exp\left[\log\,\pi\left(\boldsymbol{\beta}_{jk}^{(t)}|\cdot\right) + \left(\boldsymbol{\beta}_{jk}^{\star} - \boldsymbol{\beta}_{jk}^{(t)}\right)^{\top}\mathbf{s}\left(\boldsymbol{\beta}_{jk}^{(t)}\right) + \right. \\
&\qquad \left. \frac{1}{2}\left(\boldsymbol{\beta}_{jk}^{\star} - \boldsymbol{\beta}_{jk}^{(t)}\right)^{\top}\mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)\left(\boldsymbol{\beta}_{jk}^{\star} - \boldsymbol{\beta}_{jk}^{(t)}\right)\right] \\
&\propto \exp\left[(\boldsymbol{\beta}_{jk}^{\star})^{\top}\mathbf{s}\left(\boldsymbol{\beta}_{jk}^{(t)}\right) + \left(\frac{1}{2}(\boldsymbol{\beta}_{jk}^{\star})^{\top}\mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right) - \right. \right. \\
&\qquad \left. \left. \frac{1}{2}(\boldsymbol{\beta}_{jk}^{(t)})^{\top}\mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)\right)\left(\boldsymbol{\beta}_{jk}^{\star} - \boldsymbol{\beta}_{jk}^{(t)}\right)\right] \\
&\propto \exp\left[(\boldsymbol{\beta}_{jk}^{\star})^{\top}\mathbf{s}\left(\boldsymbol{\beta}_{jk}^{(t)}\right) + \frac{1}{2}(\boldsymbol{\beta}_{jk}^{\star})^{\top}\mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)\boldsymbol{\beta}_{jk}^{\star} - \right. \\
&\qquad \left. \frac{1}{2}(\boldsymbol{\beta}_{jk}^{\star})^{\top}\mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)\boldsymbol{\beta}_{jk}^{(t)} - \frac{1}{2}(\boldsymbol{\beta}_{jk}^{(t)})^{\top}\mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)\boldsymbol{\beta}_{jk}^{\star}\right] \\
&= \exp\left[\frac{1}{2}(\boldsymbol{\beta}_{jk}^{\star})^{\top}\mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)\boldsymbol{\beta}_{jk}^{\star} + (\boldsymbol{\beta}_{jk}^{\star})^{\top}\mathbf{s}\left(\boldsymbol{\beta}_{jk}^{(t)}\right) - (\boldsymbol{\beta}_{jk}^{\star})^{\top}\mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)\boldsymbol{\beta}_{jk}^{(t)}\right] \\
&= \exp\left[-\frac{1}{2}(\boldsymbol{\beta}_{jk}^{\star})^{\top} - \mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)\boldsymbol{\beta}_{jk}^{\star} + (\boldsymbol{\beta}_{jk}^{\star})^{\top}\left(\mathbf{s}\left(\boldsymbol{\beta}_{jk}^{(t)}\right) - \mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)\boldsymbol{\beta}_{jk}^{(t)}\right)\right]
\end{aligned}$$

Which leads to the proposal density $q(\boldsymbol{\beta}_{jk}^{\star}|\,\boldsymbol{\beta}_{jk}^{(t)}) = \mathcal{N}(\boldsymbol{\mu}_{jk}^{(t)}, \boldsymbol{\Sigma}_{jk}^{(t)})$ with precision matrix

$$\left(\boldsymbol{\Sigma}_{jk}^{(t)}\right)^{-1} = -\mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)$$

and mean

$$\begin{aligned}
\boldsymbol{\mu}_{jk}^{(t)} &= \boldsymbol{\Sigma}_{jk}^{(t)}\left[s\left(\boldsymbol{\beta}_{jk}^{(t)}\right) - \mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)\boldsymbol{\beta}_{jk}^{(t)}\right] \\
&= \boldsymbol{\beta}_{jk}^{(t)} - \mathbf{H}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right)^{-1}\mathbf{s}\left(\boldsymbol{\beta}_{jk}^{(t)}\right) \\
&= \boldsymbol{\beta}_{jk}^{(t)} - \left[\mathbf{J}_{kk}\left(\boldsymbol{\beta}_{jk}^{(t)}\right) + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}))\right]^{-1}\mathbf{s}\left(\boldsymbol{\beta}_{jk}^{(t)}\right).
\end{aligned}$$

Using a basis function representation of functions $f_{jk}(\cdot)$ the precision matrix is

$$\left(\boldsymbol{\Sigma}_{jk}^{(t)}\right)^{-1} = \mathbf{X}_{jk}^{\top}\mathbf{W}_{kk}\mathbf{X}_{jk} + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}),$$

with weights $\mathbf{W}_{kk} = -\text{diag}(\partial^2 \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X})/\partial \boldsymbol{\eta}_k \partial \boldsymbol{\eta}_k^\top)$ and the mean can be written as

$$\begin{aligned}
\boldsymbol{\mu}_{jk}^{(t)} &= \boldsymbol{\Sigma}_{jk}^{(t)} \left[\mathbf{X}_{jk}^\top \mathbf{u}_k^{(t)} - \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk}) \boldsymbol{\beta}_{jk}^{(t)} + (\mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} + \mathbf{G}_{jk}(\boldsymbol{\tau}_{jk})) \boldsymbol{\beta}_{jk}^{(t)} \right] \\
&= \boldsymbol{\Sigma}_{jk}^{(t)} \left[\mathbf{X}_{jk}^\top \mathbf{u}_k^{(t)} + \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \mathbf{X}_{jk} \boldsymbol{\beta}_{jk}^{(t)} \right] \\
&= \boldsymbol{\Sigma}_{jk}^{(t)} \left[\mathbf{X}_{jk}^\top \mathbf{u}_k^{(t)} + \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \left(\boldsymbol{\eta}_k^{(t)} - \boldsymbol{\eta}_{k,-j}^{(t)} \right) \right] \\
&= \boldsymbol{\Sigma}_{jk}^{(t)} \mathbf{X}_{jk}^\top \left[\mathbf{u}_k^{(t)} + \mathbf{W}_{kk} \left(\boldsymbol{\eta}_k^{(t)} - \boldsymbol{\eta}_{k,-j}^{(t)} \right) \right] \\
&= \boldsymbol{\Sigma}_{jk}^{(t)} \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \left[\boldsymbol{\eta}_k^{(t)} + \mathbf{W}_{kk}^{-1} \mathbf{u}_k^{(t)} - \boldsymbol{\eta}_{k,-j}^{(t)} \right] \\
&= \boldsymbol{\Sigma}_{jk}^{(t)} \mathbf{X}_{jk}^\top \mathbf{W}_{kk} \left[\mathbf{z}_k - \boldsymbol{\eta}_{k,-j}^{(t)} \right]
\end{aligned}$$

with working observations $\mathbf{z}_k = \boldsymbol{\eta}_k^{(t)} + \mathbf{W}_{kk}^{-1} \mathbf{u}_k^{(t)}$.

Affiliation:

Nikolaus Umlauf, Achim Zeileis
Department of Statistics
Faculty of Economics and Statistics
Universität Innsbruck
Universitätsstr. 15
6020 Innsbruck, Austria
E-mail: Nikolaus.Umlauf@uibk.ac.at, Achim.Zeileis@R-project.org
URL: https://eeecon.uibk.ac.at/~umlauf/, https://eeecon.uibk.ac.at/~zeileis/

Nadja Klein
Chairs of Statistics and Econometrics
Universität Göttingen
Humboldtallee 3
37073 Göttingen, Germany
E-mail: n.klein@uni-goettingen.de
URL: https://www.uni-goettingen.de/de/325353.html

University of Innsbruck - Working Papers in Economics and Statistics
Recent Papers can be accessed on the following webpage:

http://eeecon.uibk.ac.at/wopec/

2017-05 **Nikolaus Umlauf, Nadja Klein, Achim Zeileis:** BAMLSS: Bayesian additive models for location, scale and shape (and beyond)

2017-04 **Martin Halla, Susanne Pech, Martina Zweimüller:** The effect of statutory sick-pay on workers' labor supply and subsequent health

2017-03 **Franz Buscha, Daniel Müller, Lionel Page:** Can a common currency foster a shared social identity across different nations? The case of the Euro.

2017-02 **Daniel Müller:** The anatomy of distributional preferences with group identity

2017-01 **Wolfgang Frimmel, Martin Halla, Jörg Paetzold:** The intergenerational causal effect of tax evasion: Evidence from the commuter tax allowance in Austria

2016-33 **Alexander Razen, Stefan Lang, Judith Santer:** Estimation of spatially correlated random scaling factors based on Markov random field priors

2016-32 **Meike Köhler, Nikolaus Umlauf, Andreas Beyerlein, Christiane Winkler, Anette-Gabriele Ziegler, Sonja Greven:** Flexible Bayesian additive joint models with an application to type 1 diabetes research

2016-31 **Markus Dabernig, Georg J. Mayr, Jakob W. Messner, Achim Zeileis:** Simultaneous ensemble post-processing for multiple lead times with standardized anomalies

2016-30 **Alexander Razen, Stefan Lang:** Random scaling factors in Bayesian distributional regression models with an application to real estate data

2016-29 **Glenn Dutcher, Daniela Glätzle-Rützler, Dmitry Ryvkin:** Don't hate the player, hate the game: Uncovering the foundations of cheating in contests

2016-28 **Manuel Gebetsberger, Jakob W. Messner, Georg J. Mayr, Achim Zeileis:** Tricks for improving non-homogeneous regression for probabilistic precipitation forecasts: Perfect predictions, heavy tails, and link functions

2016-27 **Michael Razen, Matthias Stefan:** Greed: Taking a deadly sin to the lab

2016-26 **Florian Wickelmaier, Achim Zeileis:** Using recursive partitioning to account for parameter heterogeneity in multinomial processing tree models

2016-25 **Michel Philipp, Carolin Strobl, Jimmy de la Torre, Achim Zeileis:** On the estimation of standard errors in cognitive diagnosis models

2016-24 **Florian Lindner, Julia Rose:** No need for more time: Intertemporal allocation decisions under time pressure

2016-23 **Christoph Eder, Martin Halla:** The long-lasting shadow of the allied occupation of Austria on its spatial equilibrium

2016-22 **Christoph Eder:** Missing men: World War II casualties and structural change

2016-21 **Reto Stauffer, Jakob Messner, Georg J. Mayr, Nikolaus Umlauf, Achim Zeileis:** Ensemble post-processing of daily precipitation sums over complex terrain using censored high-resolution standardized anomalies

2016-20 **Christina Bannier, Eberhard Feess, Natalie Packham, Markus Walzl:** Incentive schemes, private information and the double-edged role of competition for agents

2016-19 **Martin Geiger, Richard Hule:** Correlation and coordination risk

2016-18 **Yola Engler, Rudolf Kerschbamer, Lionel Page:** Why did he do that? Using counterfactuals to study the effect of intentions in extensive form games

2016-17 **Yola Engler, Rudolf Kerschbamer, Lionel Page:** Guilt-averse or reciprocal? Looking at behavioural motivations in the trust game

2016-16 **Esther Blanco, Tobias Haller, James M. Walker:** Provision of public goods: Unconditional and conditional donations from outsiders

2016-15 **Achim Zeileis, Christoph Leitner, Kurt Hornik:** Predictive bookmaker consensus model for the UEFA Euro 2016

2016-14 **Martin Halla, Harald Mayr, Gerald J. Pruckner, Pilar García-Gómez:** Cutting fertility? The effect of Cesarean deliveries on subsequent fertility and maternal labor supply

2016-13 **Wolfgang Frimmel, Martin Halla, Rudolf Winter-Ebmer:** How does parental divorce affect children's long-term outcomes?

2016-12 **Michael Kirchler, Stefan Palan:** Immaterial and monetary gifts in economic transactions. Evidence from the field

2016-11 **Michel Philipp, Achim Zeileis, Carolin Strobl:** A toolkit for stability assessment of tree-based learners

2016-10 **Loukas Balafoutas, Brent J. Davis, Matthias Sutter:** Affirmative action or just discrimination? A study on the endogenous emergence of quotas *forthcoming in Journal of Economic Behavior and Organization*

2016-09 **Loukas Balafoutas, Helena Fornwagner:** The limits of guilt

2016-08 **Markus Dabernig, Georg J. Mayr, Jakob W. Messner, Achim Zeileis:** Spatial ensemble post-processing with standardized anomalies

2016-07 **Reto Stauffer, Jakob W. Messner, Georg J. Mayr, Nikolaus Umlauf, Achim Zeileis:** Spatio-temporal precipitation climatology over complex terrain using a censored additive regression model

2016-06 **Michael Razen, Jürgen Huber, Michael Kirchler:** Cash inflow and trading horizon in asset markets

2016-05 **Ting Wang, Carolin Strobl, Achim Zeileis, Edgar C. Merkle:** Score-based tests of differential item functioning in the two-parameter model

2016-04 **Jakob W. Messner, Georg J. Mayr, Achim Zeileis:** Non-homogeneous boosting for predictor selection in ensemble post-processing

2016-03 **Dietmar Fehr, Matthias Sutter:** Gossip and the efficiency of interactions

2016-02 **Michael Kirchler, Florian Lindner, Utz Weitzel:** Rankings and risk-taking in the finance industry

2016-01 **Sibylle Puntscher, Janette Walde, Gottfried Tappeiner:** Do methodical traps lead to wrong development strategies for welfare? A multilevel approach considering heterogeneity across industrialized and developing countries

University of Innsbruck

Working Papers in Economics and Statistics

2017-05

Nikolaus Umlauf, Nadja Klein, Achim Zeileis

BAMLSS: Bayesian additive models for location, scale and shape (and beyond)

Abstract
Bayesian analysis provides a convenient setting for the estimation of complex generalized additive regression models (GAMs). Since computational power has tremendously increased in the past decade it is now possible to tackle complicated inferential problems, e.g., with Markov chain Monte Carlo simulation, on virtually any modern computer. This is one of the reasons why Bayesian methods have become increasingly popular, leading to a number of highly specialized and optimized estimation engines and with attention shifting from conditional mean models to probabilistic distributional models capturing location, scale, shape (and other aspects) of the response distribution. In order to embed many different approaches suggested in literature and software, a unified modeling architecture for distributional GAMs is established that exploits the general structure of these models and encompasses many different response distributions, estimation techniques (posterior mode or posterior mean), and model terms (fixed, random, smooth, spatial, ...). It is shown that within this framework implementing algorithms for complex regression problems, as well as the integration of already existing software, is relatively straightforward. The usefulness is emphasized with two complex and computationally demanding application case studies: a large daily precipitation climatology based on more than 1.2 million observations from more than 50 meteorological stations, as well as a Cox model for continuous time with space-time interactions on a data set with over five thousand 'individuals'.

ISSN 1993-4378 (Print)
ISSN 1993-6885 (Online)