LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

INSTITUT FÜR STATISTIK

Fabian Scheipl

Normal-Mixture-of-Inverse-Gamma Priors for Bayesian Regularization and Model Selection in Structured Additive Regression Models

Technical Report Number 84, 2010
Department of Statistics
University of Munich

http://www.stat.uni-muenchen.de

Normal-Mixture-of-Inverse-Gamma Priors for Bayesian Regularization and Model Selection in Structured Additive Regression Models

Fabian Scheipl

September 8, 2010

In regression models with many potential predictors, choosing an appropriate subset of covariates and their interactions at the same time as determining whether linear or more flexible functional forms are required is a challenging and important task. We propose a spike-and-slab prior structure in order to include or exclude single coefficients as well as blocks of coefficients associated with factor variables, random effects or basis expansions of smooth functions. Structured additive models with this prior structure are estimated with Markov Chain Monte Carlo using a redundant multiplicative parameter expansion. We discuss shrinkage properties of the novel prior induced by the redundant parameterization, investigate its sensitivity to hyperparameter settings and compare performance of the proposed method in terms of model selection, sparsity recovery, and estimation error for Gaussian, binomial and Poisson responses on real and simulated data sets with that of component-wise boosting and other approaches.

Contents

1 Introduction

In data sets with many potential predictors, choosing an appropriate subset of covariates and their interactions at the same time as determining whether linear or more flexible functional forms are required to model the relationship between covariates and response is a challenging and important task. From a Bayesian perspective, it can be translated into a question of estimating marginal posterior probabilities whether a variable should be in the model and in what form (i.e. linear or smooth; as main effect and/or as effect modifier).

This report describes a method based on a spike-and-slab prior structure [Ishwaran and Rao, 2005] to select or deselect single coefficients as well as blocks of coefficients associated with factor variables, interactions or basis expansions of smooth functions. These bimodal priors for the hyper-variances of the regression coefficients result in a two component mixture of a narrow "spike" around zero and a "slab" with wide support as the marginal prior for the coefficients. The mixture weights for the "spike" component can be interpreted as posterior probabilities of exclusion of a coefficient or coefficient block from the model.

The main contribution of the present work is the extension of the spike-and-slab or stochastic search variable selection (SSVS) approach [George and McCulloch, 1993] for selection of single coefficients in Gaussian models to the selection of potentially large blocks of coefficients for general responses from an exponential family. We use an innovative sampling procedure based on a redundant multiplicative parameter expansion [Gelman et al., 2008] in order to improve the exceedingly slow mixing of conventional samplers that make a direct extension of the spike-and-slab approach for function selection (or, more generally, selection of coefficient blocks) infeasible. We also show that this parameter expansion leads to a prior with desirable regularization properties similar to $\mathcal{L}_q$-penalization with $q < 1$. To make our approach reproducible and applicable, it is implemented in publicly available software (R-package spikeSlabGAM [Scheipl, 2010c]). It improves on previous approaches in that it fulfills all of the following criteria simultaneously:

i. it accommodates all types of regularized effects with a (conditionally) Gaussian prior such as simple covariates (both metric and categorical), penalized splines (uni- or multivariate), random effects or ridge-penalized factors/interaction effects,

ii. it scales reasonably well to intermediate datasets with thousands of observations and hundreds of covariates,

iii. it accommodates non-Gaussian responses from the exponential family,

iv. it is implemented in publicly available and user-friendly open source software.

Fitting the practical importance of the topic, a vast literature on Bayesian approaches for selection of single coefficients based on mixture priors for the

coefficients exists. In a recent review paper, O'Hara and Sillanpää [2009] compare the spike-and-slab approach in Kuo and Mallick [1998], the Gibbs variable selection approach [Carlin and Chib, 1995, Dellaportas et al., 2002], and stochastic search variable selection (SSVS) approaches in George and McCulloch [1993], among other methods.

Bayesian function selection, similar to the frequentist COSSO [Zhang and Lin, 2003], is usually based on decomposing an additive model into orthogonal functions in the spirit of a smoothing spline ANOVA [Wahba et al., 1995]. Wood et al. [2002] and Yau et al. [2003] describe implementations using a data-based prior that requires two MCMC runs, a pilot run to obtain a data-based prior for the "slab" part and a second one to estimate parameters and select model components. A more general approach based on double exponential regression models that also allows for flexible modeling of the dispersion is described by Cottet et al. [2008]. They use a reduced rank representation of cubic smoothing splines (i.e a "pseudo-spline" [Hastie, 1996]) with a very small number of basis functions to model the smooth terms in order to reduce the complexity of the fitted models, and, presumably, to avoid the mixing problems detailed in Section 3.2. Since the authors were unable to provide their software for this work, it was not possible to compare their approach to the one described in the following. Reich et al. [2009] also use the smoothing spline ANOVA framework and perform variable and function selection via SSVS for Gaussian responses, but their implementation is very slow. To the best of our knowledge, none of the above-mentioned approaches was implemented in publicly available software in a useable form at the time of writing and none are able to select between smooth nonlinear and linear effects.

The report is structured as follows: Section 2 summarizes structured additive regression models and introduces the notation. Section 3 describes the prior structure (3.1) and the parameter expansion trick used to improve mixing (3.2) and discusses shrinkage properties of the marginal prior for the regression coefficients (3.4). Section 4 describes the MCMC sampler implemented in spikeSlabGAM. Sections 5 and 6 summarize results from a variety of simulation studies and a collection of real data sets, respectively.

2 Structured additive regression

2.1 Model structure

Structured additive regression [Fahrmeir et al., 2004], a broad model class that contains generalized additive mixed models, is among the most widely used approaches in applied statistics due to its flexibility and generality.

We give a short summary of structured additive regression: The distribution of the responses $\boldsymbol{y}$ given a set of covariates $\boldsymbol{x}_j$; $j = 1, \ldots, p$ belongs to an exponential family, i.e

$$\pi(y|x, \phi) = c(y, \phi) \exp\left(\frac{y\theta - b(\theta)}{\phi}\right), \tag{1}$$

with $\theta, \phi, b(\cdot)$ and $c(\cdot)$ determined by the type of distribution. The additive predictor $\boldsymbol{\eta} = \sum_{j=1}^{p} f_j(\boldsymbol{x}_j)$ determines the conditional expected value of the response via

$$\mathrm{E}(\boldsymbol{y}|\boldsymbol{x}_{1,\ldots,p}) = h(\boldsymbol{\eta}) \tag{2}$$

with a fixed response function $h(\cdot)$.

Components $f(x)$ of the additive predictor can contain a wide variety of regularized and unregularized model terms, such as

- linear terms $f(x) = \beta x$
- factor variables ($f(x) = \beta_{x(i)}$ iff $x = i$)
- interactions (both linear-linear or categorical-linear)
- smooth functions of (one or more) continuous covariates, i.e. splines, spatial effects, surface estimators, varying coefficient terms
- Gaussian Markov random fields for discrete spatial covariates
- random effects such as subject-specific intercepts.

Flexible terms such as the last 3 need to be regularized in order to avoid overfitting and are modeled with appropriate shrinkage priors. These shrinkage or regularization priors are usually Gaussian or can be parameterized as scale mixtures of Gaussians (e.g. the Bayesian Lasso with a Laplace prior on the coefficients is a Normal-Exponential scale mixture [Park and Casella, 2008]), so that they are conditionally Gaussian given their variance parameters. In the following we focus on models including linear terms, factor variables, smooth functions of a single covariate and random intercept terms.

2.2 Bayesian P-splines

Smooth functions $f()$ of continuous covariates are commonly modeled via basis function expansions, i.e. $f(\boldsymbol{x}) = \sum_{k=1}^{K} \delta_k B_k(\boldsymbol{x}) = \boldsymbol{B}\boldsymbol{\delta}$, where $\boldsymbol{\delta}$ is a vector of coefficients associated with (nonlinear) basis functions $B_k(); k = 1, \ldots, K$. Many possibilities for the choice of basis functions and the associated regularization exist. Knot-free methods include e.g. thin plate splines [Wood, 2003] or smoothing splines [Wood et al., 2002] and their reduced rank representations [Cottet et al., 2008] based on the dominating eigenvalues and -vectors of the covariance of the equivalent Gaussian process.

In the following, we use Bayesian P-splines as introduced by Lang and Brezger [2004], similar to the approach chosen in Panagiotelis and Smith [2008]. In this approach, $B_k(x), k = 1, \ldots, K$ is a collection of B-spline basis functions [Eilers and Marx, 1996] and the shrinkage prior on the associated coefficient vector $\boldsymbol{\delta}$ is a Gaussian random walk prior of order d:

$$\Delta^d \boldsymbol{\delta} \sim N_{K-d}\left(\mathbf{0}, \tau^2 \boldsymbol{I}_{K-d}\right),$$

where Δ^d is the d-th difference operator matrix. In the following we use cubic B-splines with a second order difference penalty. Note that this formulation implies a partially improper prior for δ: $\delta \propto \exp\left(-0.5\delta' \boldsymbol{P}\delta/\tau^2\right)$, with rank-deficient $\boldsymbol{P} = \Delta^{d'}\Delta^d$.

2.3 Decomposition and Reparameterization of regularized terms in penalized and unpenalized parts

For both computational and interpretational reasons it is often beneficial to reparameterize regularized model components with a partially improper prior in a mixed model representation [Fahrmeir et al., 2004]. Partially improper priors naturally arise e.g. for P-splines because the prior is a Bayesian analogue to the frequentist roughness penalty which is constructed so that, for d-th order differences, polynomial functions up to the $(d-1)$-th power remain unpenalized. Consequently, coefficient vectors that parameterize constant or linear functions are in the nullspace of the prior precision matrix.

More generally, for any regularized term $f(\boldsymbol{x}) = \boldsymbol{B}\delta$ with a partially improper Gaussian prior $\delta \sim N_K\left(\boldsymbol{0}, s^2\boldsymbol{P}^-\right)$ with fixed rank-deficient precision matrix $\boldsymbol{P}$ and associated design matrix $\boldsymbol{B}$, the problem is reparameterized by a decomposition of the coefficient vector δ into an unpenalized part and a penalized part:

$$\delta = \tilde{\boldsymbol{X}}_{\boldsymbol{1}}\boldsymbol{\beta}_1 + \tilde{\boldsymbol{X}}_{\boldsymbol{2}}\boldsymbol{\beta}_2$$

where $\tilde{\boldsymbol{X}}_{\boldsymbol{1}} \in \mathbb{R}^{K\times d}$, is a basis of the d-dimensional nullspace of $\boldsymbol{P}$ and $\tilde{\boldsymbol{X}}_{\boldsymbol{1}}$ and $\tilde{\boldsymbol{X}}_{\boldsymbol{2}}$ have the following properties [Kneib, 2006, ch. 5.1]:

1. The concatenated matrix $[\tilde{\boldsymbol{X}}_{\boldsymbol{1}}\tilde{\boldsymbol{X}}_{\boldsymbol{2}}]$ has full rank to make the transformation above a one-to-one transformation. This also implies that both $\tilde{\boldsymbol{X}}_{\boldsymbol{1}}$ and $\tilde{\boldsymbol{X}}_{\boldsymbol{2}}$ have full column rank.
2. $\tilde{\boldsymbol{X}}_{\boldsymbol{1}}$ and $\tilde{\boldsymbol{X}}_{\boldsymbol{2}}$ are orthogonal, i. e. $\tilde{\boldsymbol{X}}_{\boldsymbol{1}}\tilde{\boldsymbol{X}}_{\boldsymbol{2}}{}' = \boldsymbol{0}$
3. $\tilde{\boldsymbol{X}}_{\boldsymbol{1}}'\boldsymbol{P}\tilde{\boldsymbol{X}}_{\boldsymbol{1}} = \boldsymbol{0}$, so that $\boldsymbol{\beta}_1$ is unpenalized by $\boldsymbol{P}$
4. $\tilde{\boldsymbol{X}}_{\boldsymbol{2}}'\boldsymbol{P}\tilde{\boldsymbol{X}}_{\boldsymbol{2}} = \boldsymbol{I}$, so that the penalty term for $\boldsymbol{\beta}_2$ reduces to $\|\boldsymbol{\beta}_2\|^2$, the kernel of a vector of *i.i.d.* Gaussian variates.

The decomposition is not unique and can always be based on the spectral decomposition of $\boldsymbol{P}$. With

$$\boldsymbol{P} = [\boldsymbol{\Lambda}_+\boldsymbol{\Lambda}_{\boldsymbol{0}}]' \begin{pmatrix} \boldsymbol{\Gamma}_+ & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix} [\boldsymbol{\Lambda}_+\boldsymbol{\Lambda}_{\boldsymbol{0}}],$$

where $\boldsymbol{\Lambda}_+$ is the matrix of the eigenvectors associated with the positive eigenvalues $\mathrm{diag}(\boldsymbol{\Gamma}_+)$, and $\boldsymbol{\Lambda}_0$ are the eigenvectors associated with the zero eigenvalues, the decomposition is

$\tilde{\boldsymbol{X}}_{\boldsymbol{1}} = \boldsymbol{\Lambda}_0$ and

$\tilde{\boldsymbol{X}}_{\boldsymbol{2}} = \boldsymbol{L}(\boldsymbol{L}'\boldsymbol{L})^{-1}$ with $\boldsymbol{L} = \boldsymbol{\Lambda}_+{\boldsymbol{\Gamma}_+}^{1/2}$.

The regularized model terms can then be expressed as

$$\begin{aligned} \boldsymbol{B\delta} &= \boldsymbol{B}(\tilde{\boldsymbol{X}}_{\boldsymbol{1}}\boldsymbol{\beta}_{\boldsymbol{1}} + \tilde{\boldsymbol{X}}_{\boldsymbol{2}}\boldsymbol{\beta}_{\boldsymbol{2}}) = \boldsymbol{X}_{\boldsymbol{1}}\boldsymbol{\beta}_{\boldsymbol{1}} + \boldsymbol{X}_{\boldsymbol{2}}\boldsymbol{\beta}_{\boldsymbol{2}} \\ \text{and} \quad \boldsymbol{\delta}'\boldsymbol{P}\boldsymbol{\delta} &= (\tilde{\boldsymbol{X}}_{\boldsymbol{1}}\boldsymbol{\beta}_{\boldsymbol{1}} + \tilde{\boldsymbol{X}}_{\boldsymbol{2}}\boldsymbol{\beta}_{\boldsymbol{2}})'\boldsymbol{P}(\tilde{\boldsymbol{X}}_{\boldsymbol{1}}\boldsymbol{\beta}_{\boldsymbol{1}} + \tilde{\boldsymbol{X}}_{\boldsymbol{2}}\boldsymbol{\beta}_{\boldsymbol{2}}) = \boldsymbol{\beta}_{\boldsymbol{2}}'\boldsymbol{\beta}_{\boldsymbol{2}} \end{aligned} \tag{3}$$

with $\boldsymbol{X}_1$ as the design matrix associated with the unpenalized part and $\boldsymbol{X}_2$ as the design matrix associated with the penalized part of the term. The prior for the regularized part after reparameterization is then $\boldsymbol{\beta}_2 \sim N_{K-d}(\boldsymbol{0}, s^2\boldsymbol{I})$, while $\boldsymbol{\beta}_1$ has a flat prior. For an additive model with linear predictor $\boldsymbol{\eta}$ given by $\boldsymbol{\eta} = \sum_{k=1}^{p} f_k(\boldsymbol{x_k}) = \sum_{k=1}^{p} \boldsymbol{B_k}\boldsymbol{\delta_k}$, the reparameterization results in a linear predictor $\boldsymbol{\eta} = \sum_{k=1}^{p} \boldsymbol{X_{k,1}}\boldsymbol{\beta_{k,1}} + \sum_{k=1}^{p} \boldsymbol{X_{k,2}}\boldsymbol{\beta_{k,2}}$.

3 The NMIG Model with Parameter Expansion

The following Section describes the prior structure of the conventional Normal-mixture of Inverse Gamma (NMIG) model (Section 3.1) and shows that this setup is not well suited for the simultaneous selection of coefficient groups (Section 3.2). Section 3.3 describes a parameter expansion that changes the prior structure and enables simultaneous selection of coefficient groups. Ishwaran and Rao [2005] originally proposed an empirical Bayes analogue of this prior for selection of single coefficients in the linear model for Gaussian data.

3.1 Model Hierarchy

This section discusses the basic model hierarchy for structured additive regression models with the NMIG prior. In most cases, the linear predictor $\boldsymbol{\eta}$ will contain terms that are forced into the model (e.g. a global intercept term) and are not associated with a variable selection prior. We write $\boldsymbol{\eta} = \boldsymbol{\eta}_u + \boldsymbol{X\beta}$, where $\boldsymbol{\eta}_u = \boldsymbol{X}_u\boldsymbol{\beta}_u$ represents the part of the linear predictor not associated with an NMIG prior. In the following, we focus on the part $\boldsymbol{X\beta}$ associated with NMIG priors.

Figure 1 shows the hierarchy of the basic NMIG prior model. At the lowest level of the hierarchy, the data y_i, $i = 1, \ldots, n$ come from a distribution in the exponential family such as the Gaussian, binomial or poisson distributions. The canonical parameter of this distribution is connected to the linear predictor via a known response function $g()$. The regression coefficients have independent Gaussian priors with mean zero. Subvectors β_j, $j = 1, \ldots, p$ are associated with different components of the predictor, i.e. different covariates, unpenalized and penalized parts of a reparameterized spline basis or a set of indicator variables encoding the levels of a factor. The prior variance for β is constant within subvectors and given by the product of an indicator variable γ_j and the hypervariance τ_j^2. The indicator variable γ_j takes the value 1 with probability w or some (very) small value v_0 with probability $1 - w$. The hypervariance τ_j^2 has an inverse gamma-prior with shape parameter a_τ

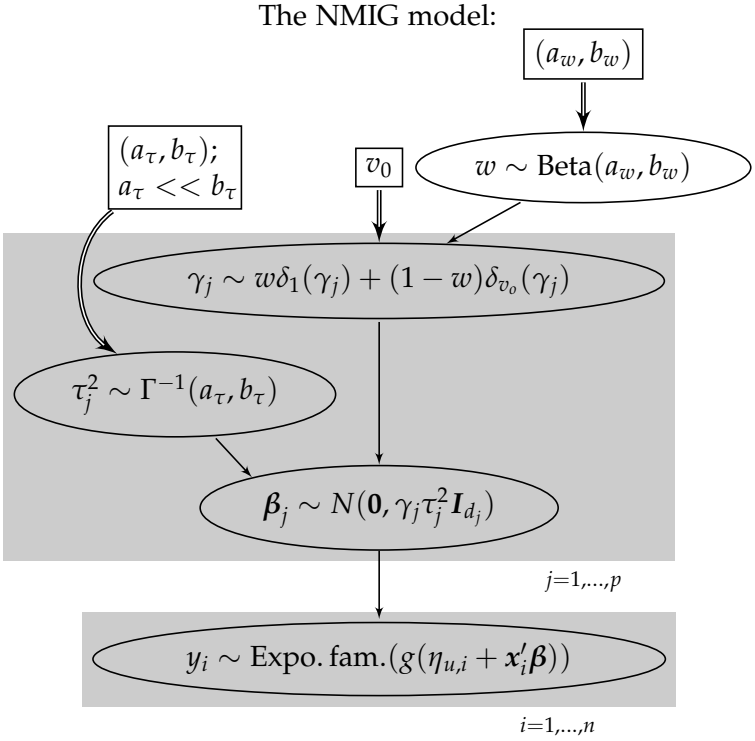

Figure 1: Directed acyclic graph for the NMIG model.
Ellipses are stochastic nodes, rectangles are deterministic/logical nodes. Single arrows are stochastic edges, double arrows are logical/deterministic edges. Subvectors $\boldsymbol{\beta}_j$ are associated with different components of the predictor, i.e. unpenalized and penalized parts of a reparameterized spline basis or indicators coding the different levels of a factor. d_j is the length of subvector $\boldsymbol{\beta}_j$. $g()$ is a known response function. $\delta_y(x)$ is zero for any value of x other than y and 1 at y. The linear predictor from model terms not associated with an NMIG prior is given by $\eta_{u,i}$.

and scale parameter b_τ with $b_\tau \gg a_\tau$, so that the mode b_τ / a_τ is significantly greater than 1. The implied prior for the effective hypervariance $v_j^2 = \gamma_j \tau_j^2$ is a bimodal mixture of inverse gamma distributions, with one component strongly concentrated on very small values – the *spike* with $\gamma_j = v_0$ and effective scale parameter $v_0 b_\tau$ – and a second more diffuse component with most mass on larger values – the *slab* with $\gamma_j = 1$ and scale b_τ. A coefficient associated with a hypervariance that is primarily sampled from the *spike*-part of the prior will be strongly shrunk towards zero if v_0 is sufficiently small, so that the posterior probability for $\gamma_j = v_0$ can be interpreted as the probability of exclusion of $\boldsymbol{\beta}_j$ from the model. The Beta prior for the mixture weights w can be used to incorporate the analyst's prior knowledge about the sparsity of $\boldsymbol{\beta}$ or, more practically, enforce sufficiently sparse solutions for overparameterized models. In the following, we write $\boldsymbol{\beta}_j \sim \text{NMIG}(v_0, w, a_\tau, b_\tau)$ to denote this prior hierarchy for the regression coefficients.

Expressions for the full conditionals resulting from this prior structure are given in Section 4. This prior hierarchy is very well suited for selection of model terms for non-Gaussian data because the selection (i.e. the sampling of indicator variables γ) occurs on the level of the hypervariances for the coefficients. This means that the likelihood itself is not in the Markov blanket of γ and consequently does not occur in the full conditionals for the indicator variables. Since the full conditionals for γ are thus available in closed form regardless of the likelihood, this results in comparatively easy and fast model averaging for non-Gaussian models without the need to delve into the intricacies of estimating marginal likelihoods.

3.2 Using NMIG for simultaneous selection of multiple coefficients fails

Previous approaches for Bayesian variable selection have primarily concentrated on selection of single coefficients [George and McCulloch, 1993, Kuo and Mallick, 1998, Dellaportas et al., 2002, Ishwaran and Rao, 2005] or used very low dimensional bases for the representation of smooth effects. E.g. Cottet et al. [2008] use a pseudo-spline representation of their cubic smoothing spline bases with only 3 to 4 basis functions. In the following, we argue that conventional blockwise Gibbs sampling is ill suited for updating the state of the Markov chain when sampling from the posterior of an NMIG model even for moderately large coefficient blocks. We show that mixing for γ_j will be very slow for blocks of coefficients $\boldsymbol{\beta}_j$ with $d_j \gg 1$. We suppress the index j in the following.

The following analysis will show that, even if the blockwise sampler is initially in an ideal state for switching between the spike and the slab parts of the prior, i.e. a parameter constellation so that the full conditional probability $P(\gamma = 1|\cdot) = .5$, such a switch is very unlikely in subsequent iterations for coefficient vectors with more than a few entries given the NMIG prior hierarchy.

Assume that the sampler starts out in iteration (0) with a parameter con-

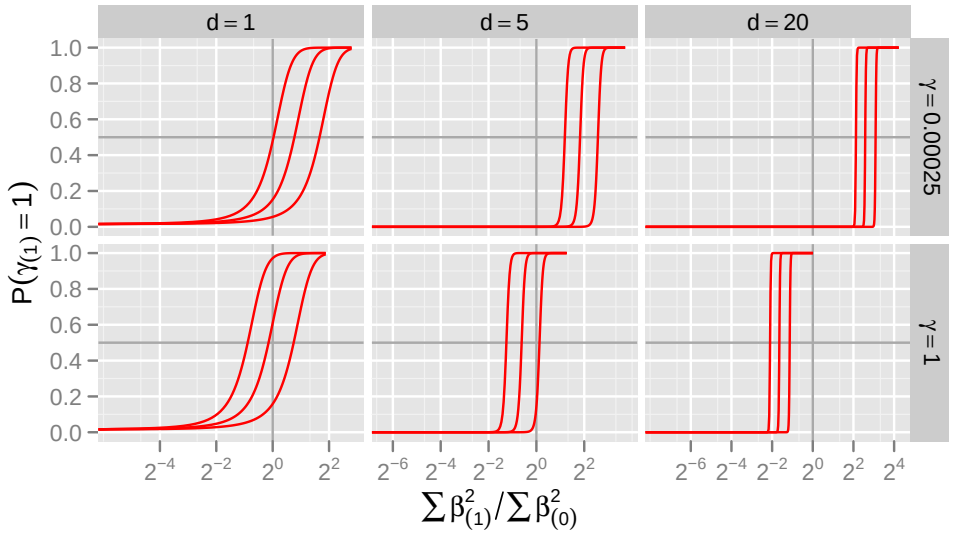

Figure 2: $P(\gamma)$ as a function of the relative change in $\sum^d \beta^2$ for varying $d, \gamma_{(0)}$: Inclusion probability in iteration (1) as a function of the ratio between the sum of squared coefficients in iteration (1) and (0). Lines in each panel correspond to $\tau^2_{(1)}$ equal to the median of its full conditional and the .1- and .9-quantiles. Upper row is for $\gamma_{(0)} = 1$, lower row for $\gamma_{(0)} = v_0$. Columns correspond to $d = 1,\ 5,\ 20$. Solid gray grid lines denote inclusion probability $= .5$ and ratio of coefficient sum of squares $= 1$

figuration of $a_t, b_t, v_0, w, \tau^2_{(0)}$ and $\beta_{(0)}$ so that $P(\gamma_{(0)} = 1|\cdot) = .5$. We set $w = .5$. The parameters for which $P(\gamma = 1|\cdot) = .5$ satisfy the following relations:

$$\frac{P(\gamma = 1|\cdot)}{P(\gamma = v_0|\cdot)} = v_0^{d/2} \exp\left(\frac{(1 - v_0)}{2v_0} \frac{\sum^d \beta^2}{\tau^2}\right) = 1,$$

so that $P(\gamma = 1|\cdot) > .5$ if

$$\begin{aligned} \frac{\sum^d \beta^2}{d\tau^2} &> -\frac{v_0}{1 - v_0} \log(v_0), \\ \text{or } \sum^d \beta^2 &> -\frac{dv_0}{1 - v_0} \log(v_0)\tau^2, \\ \text{or } \tau^2 &> -\frac{(1 - v_0) \sum^d \beta^2}{dv_0 \log(v_0)}. \end{aligned}$$

Assuming a given value $\tau^2_{(0)}$, set

$$\sum^d \beta^2_{(0)} = \frac{dv_0}{1 - v_0} \log(v_0)\tau^2_{(0)}.$$

Now $\gamma_{(0)}$ takes on both values v_0 and 1 with equal probability, conditional on all other parameters.

In the following iteration, $\tau^2_{(1)}$ is drawn from its full conditional $\Gamma^{-1}(a_t + d/2, b_t + \frac{\sum^d \beta^2_{(0)}}{2\gamma_{(0)}})$ (see (7)). Figure 2 shows $P(\gamma_{(1)} = 1|\tau^2_{(1)}, \sum^d \beta^2_{(1)})$ as a function of $\sum^d \beta^2_{(1)} / \sum^d \beta^2_{(0)}$ for various values of d. The 3 lines in each panel correspond to $P(\gamma_{(1)} = 1|\tau^2_{(1)}, \sum^d \beta^2_{(1)})$ for values of $\tau^2_{(1)}$ equal to the median of its full conditional as well as the .1- and .9-quantiles. The upper row in the Figure plots the function for $\gamma_{(0)} = 1$, the lower row for $\gamma_{(0)} = v_0$.

So, if we start in this "equilibrium state" we begin iteration (0) with v_0, w, $\tau^2_{(0)}$, and $\boldsymbol{\beta}_{(0)}$ so that $P(\gamma_{(0)} = 1|\cdot) = .5$. We then determine $P(\gamma_{(1)} = 1|\tau^2_{(1)}, \sum^d \beta^2_{(1)})$ as a function of $\sum^d \beta^2_{(1)} / \sum^d \beta^2_{(0)}$ for

- various values of $\dim(\boldsymbol{\beta}_j) = d$,
- $\gamma_{(0)} = 1$ and $\gamma_{(0)} = v_0$,
- $\tau^2_{(1)}$ at the .1, .5, .9-quantiles of its conditional distribution given $\boldsymbol{\beta}_{(0)}, \gamma_{(0)}$.

The leftmost column in Figure 2 shows that moving between $\gamma = 1$ and $\gamma = v_0$ is easy for $d = 1$: For a large range of realistic values for $\sum^d \beta^2_{(1)} / \sum^d \beta^2_{(0)}$, moving back to $\gamma_{(1)} = v_0$ from $\gamma_{(0)} = 1$ (upper panel) has reasonably large probability, just as moving from $\gamma_{(0)} = v_0$ to $\gamma_{(1)} = 1$ (lower panel) is fairly likely for realistic values of $\sum^d \beta^2_{(1)} / \sum^d \beta^2_{(0)}$. For $d = 5$, however, $P(\gamma_{(1)} = 1|\cdot)$ already resembles a step function. For $d = 20$, if $\sum^d \beta^2_{(1)} / \sum^d \beta^2_{(0)}$ is not smaller than 0.48, the probability of moving from $\gamma_{(0)} = 1$ to $\gamma_{(1)} = v_0$ (upper panel) is practically zero for 90% of the values drawn from $p(\tau^2_{(1)}|\cdot)$. However, draws of β that reduce $\sum^d \beta^2$ by more than a factor of 0.48 while $\gamma = 1$ are unlikely to occur in real data. It is also extremely unlikely to move back to $\gamma_{(1)} = 1$ when $\gamma_{(0)} = v_0$, unless $\sum^d \beta^2_{(1)} / \sum^d \beta^2_{(0)}$ is larger than 2.9. Since the full conditional for β is very concentrated if $\gamma = v_0$, such moves are highly improbable and correspondingly the sampler is unlikely to move away from $\gamma = v_0$. Numerical values for the graphs in Figure 2 were computed for $a_\tau = 5$, $b_\tau = 50$, $v_0 = 0.005$ but similar problems arise for all suitable hyperparameter configurations.

In summary, mixing of the indicator variables γ will be very slow for long subvectors. In experiments, we observed posterior means of $P(\gamma = 1)$ to be either ≈ 0 or ≈ 1 across a wide variety of settings, even for very long chains, largely depending on the starting values of the chains. The following section describes a possible remedy.

3.3 Parameter Expansion: The peNMIG Model

The mixing problem analyzed in the previous section is similar to the mixing problems encountered in other samplers for hypervariances of regression coefficients: a small variance for a batch of coefficients implies small coefficient values and small coefficient values in turn imply a small variance so that the sampler is unlikely to exit a basin of attraction around the origin. This

problem has been previously described in Gelman et al. [2008], where the issue is framed as one of strong dependence between a block of coefficients and their associated hypervariance. A bimodal prior for the variance such as the NMIG prior where the Markov chain must switch between the different components of the mixture prior associated with the two modes of course exacerbates these difficulties. A promising strategy to reduce this dependence is the introduction of working parameters that are only partially identifiable along the lines of *parameter expansion* or *marginal augmentation* introduced for the EM-algorithm in Meng and van Dyk [1997] and developed further for Bayesian inference for hierarchical models in Gelman et al. [2008]. While Gelman et al. [2008] concentrate on speeding up convergence for conventional hierarchical models, we use the parameter expansion to enable simultaneous selection or deselection of coefficient subvectors and improve the shrinkage properties of the resulting marginal prior.

We add a redundant multiplicative parameterization to the spike-and-slab prior. We set

$$\boldsymbol{\beta}_j = \alpha_j \boldsymbol{\xi}_j; \quad \boldsymbol{\xi}_j \in \mathbb{R}^{d_j}$$

for a subvector $\boldsymbol{\beta}_j$ with length d_j and use a scalar parameter

$$\alpha_j \sim \text{NMIG}(v_0, w, a_\tau, b_\tau),$$

where NMIG denotes the prior hierarchy given in Fig. 1. Entries of the vector $\boldsymbol{\xi}_j$ are a priori distributed as

$$\xi_{jk} \overset{\text{i.i.d.}}{\sim} \frac{1}{2}N(1,1) + \frac{1}{2}N(-1,1),\ k = 1, \ldots, d_j,$$

and prior independence between α_j and $\boldsymbol{\xi}_j$. We write

$$\boldsymbol{\beta}_j \sim \text{peNMIG}(v_0, w, a_\tau, b_\tau)$$

as shorthand for this prior structure.

The effective dimension of the coefficient vector associated with updating γ_j and τ_j^2 is then equal to one in every penalization group, since the Markov blankets of both γ_j and τ_j now only contain the scalar parameter α_j instead of the vector $\boldsymbol{\beta}_j$. This is crucial in order to avoid the mixing problems described in the previous Section, because instead of

$$\frac{P(\gamma = 1|\cdot)}{P(\gamma = v_0|\cdot)} = v_0^{d/2} \exp\left(\frac{(1-v_0)}{2v_0}\frac{\sum_i^d \beta_i^2}{\tau^2}\right)$$

for the conventional NMIG prior, we now have

$$\frac{P(\gamma = 1|\cdot)}{P(\gamma = v_0|\cdot)} = \sqrt{v_0} \exp\left(\frac{(1-v_0)}{2v_0}\frac{\alpha^2}{\tau^2}\right),$$

which is less susceptible to result in extreme values and behaves more like the probabilities in the leftmost column of Figure 2.

In our parameter expansion, the parameter α_j parameterizes the "importance" of the j-th coefficient block, while $\boldsymbol{\xi}_j$ "distributes" α_j across the entries in $\boldsymbol{\beta}_j$. Setting $\mathrm{E}(\xi) = \pm 1$ shrinks ξ towards $|1|$, the multiplicative identity, so that the interpretation of α_j as the "importance" of the j-th coefficient block can be maintained and yields a marginal prior for $\boldsymbol{\beta}_j$ that is less concentrated on small absolute values than $\xi \sim N(0, 1)$.

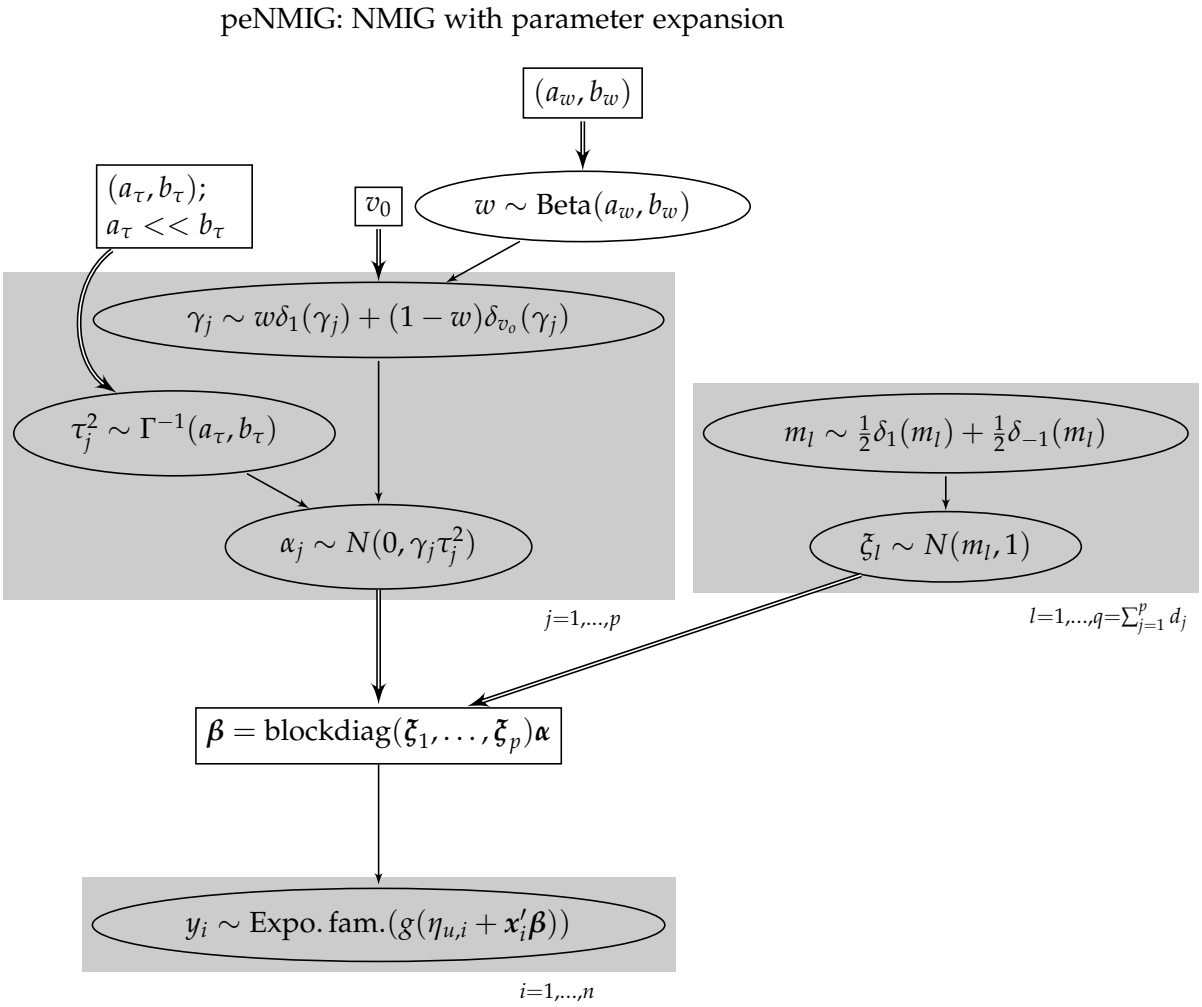

Figure 3: Directed acyclic graph of NMIG model with parameter expansion. Ellipses are stochastic nodes, rectangles are deterministic/logical nodes. Single arrows are stochastic edges, double arrows are logical/deterministic edges.

Figure 3 shows the prior hierarchy for the model with parameter expansion. In the following, this model will be denoted as peNMIG. The vector $\boldsymbol{\xi} = (\boldsymbol{\xi}_1', \ldots, \boldsymbol{\xi}_p)'$ is decomposed into subvectors $\boldsymbol{\xi}_j$ associated with the different penalization groups and their respective entries α_j, $j = 1, \ldots, p$ in $\boldsymbol{\alpha}$.

3.4 Shrinkage properties

Marginal priors

This section investigates the regularization properties of the marginal prior for the regression coefficients $\boldsymbol{\beta}$ implied by the hierarchical prior structures given in Figs. 1 and 3. To distinguish between the conventional NMIG prior and its parameter expanded version we write β if the parameter has an NMIG prior and β_{pe} if it has the parameter expanded peNMIG prior. In the following, we analyze the univariate marginal priors

$$\begin{aligned} &p(\beta|a_\tau, b_\tau, a_w, b_w, v_0) = \\ &\quad = \int p(\beta|\gamma, \tau^2) p(\tau^2|a_\tau, b_\tau) p(\gamma|w, v_0) p(w|a_w, b_w) d\tau^2 d\gamma dw \end{aligned}$$

for the conventional NMIG model and

$$\begin{aligned} &p(\beta_{pe} = \alpha\xi|a_\tau, b_\tau, a_w, b_w, v_0) \\ &\quad = \int p(\alpha|\gamma, \tau^2) \underbrace{p\left(\frac{\beta_{pe}}{\alpha}\right)}_{=\xi} \frac{1}{|\alpha|} p(\tau^2|a_\tau, b_\tau) p(\gamma|a_w, b_w, v_0) \\ &\qquad\qquad p(w|a_w, b_w) d\alpha d\tau^2 d\gamma dw \end{aligned}$$

for the peNMIG prior.

These are the univariate marginal priors for a single regression coefficient with and without parameter expansion with the intermediate quantities τ^2, γ and w integrated out. We analyze the marginal priors because it has been shown that the shrinkage properties of the resulting posterior means are dependent on their shape and less on that of the conditional priors [Fahrmeir et al., 2010, Kneib et al., 2010]. We use $v^2 = \gamma\tau^2 \sim \Gamma^{-1}(a_\tau, \gamma b_\tau)$ so that the marginal prior for β in the conventional NMIG-model is a mixture of scaled t-distributions with $2a_\tau$ degrees of freedom and scale factors $\sqrt{v_0 b_\tau / a_\tau}$ and $\sqrt{b_\tau / a_\tau}$ with weights $\frac{b_w}{a_w + b_w}$ and $\frac{a_w}{a_w + b_w}$, respectively:

$$\begin{aligned} &p(\beta|a_\tau, b_\tau, a_w, b_w, v_0) = \\ &\quad = \frac{a_w}{a_w + b_w} \int_0^\infty p(\beta|v^2) p(v^2|a_\tau, b_\tau) dv^2 \\ &\quad + \frac{b_w}{a_w + b_w} \int_0^\infty p(\beta|v^2) p(v^2|a_\tau, v_0 b_\tau) dv^2 \\ &\quad = \frac{a_w}{a_w + b_w} \frac{b_\tau^{a_\tau}}{\sqrt{2\pi}\Gamma(a_\tau)} \int_0^\infty v^{-2(a+\frac{3}{2})} e^{\left(-\frac{\frac{\beta^2}{2} + b_\tau}{v^2}\right)} dv^2 \end{aligned}$$

$$
\begin{aligned}
&+ \frac{b_w}{a_w + b_w} \frac{(v_0 b_\tau)^{a_\tau}}{\sqrt{2\pi}\Gamma(a_\tau)} \int_0^\infty v^{-2(a+\frac{3}{2})} e^{\left(-\frac{\frac{\beta^2}{2}+v_0 b_\tau}{v^2}\right)} dv^2 \\
&= K_1 \int_0^\infty \left(\frac{v^2}{\frac{\beta^2}{2}+b_\tau}\right)^{-(a+\frac{3}{2})} e^{\left(-\frac{\frac{\beta^2}{2}+b_\tau}{v^2}\right)} \left(\frac{\beta^2}{2}+b_\tau\right)^{-(a_\tau+\frac{1}{2})} d\frac{v^2}{\frac{\beta^2}{2}+b_\tau} \\
&+ K_2 \int_0^\infty \left(\frac{v^2}{\frac{\beta^2}{2}+v_0 b_\tau}\right)^{-(a_\tau+\frac{3}{2})} e^{\left(-\frac{\frac{\beta^2}{2}+v_0 b_\tau}{v^2}\right)} \left(\frac{\beta^2}{2}+v_0 b_\tau\right)^{-(a_\tau+\frac{1}{2})} \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad d\frac{v^2}{\frac{\beta^2}{2}+v_0 b_\tau} \\
&= \frac{a_w}{a_w+b_w} \frac{b_\tau^{a_\tau}\Gamma(a_\tau+\frac{1}{2})}{\sqrt{2\pi}\Gamma(a_\tau)\left(\frac{\beta^2}{2}+b_\tau\right)^{a_\tau+\frac{1}{2}}} + \frac{b_w}{a_w+b_w} \frac{(v_0 b_\tau)^{a_\tau}\Gamma(a_\tau+\frac{1}{2})}{\sqrt{2\pi}\Gamma(a_\tau)\left(\frac{\beta^2}{2}+v_0 b_\tau\right)^{a+\frac{1}{2}}} \\
&= \frac{a_w}{a_w+b_w} \frac{\Gamma\left(\frac{2a_\tau+1}{2}\right)}{\Gamma\left(\frac{2a_\tau}{2}\right)\sqrt{2a_\tau\pi\frac{b_\tau}{a_\tau}}} \left(1+\frac{\beta^2}{2a_\tau\frac{b_\tau}{a_\tau}}\right)^{-\frac{2a_\tau+1}{2}} \\
&+ \frac{b_w}{a_w+b_w} \frac{\Gamma\left(\frac{2a_\tau+1}{2}\right)}{\Gamma\left(\frac{2a_\tau}{2}\right)\sqrt{2a_\tau\pi\frac{v_0 b_\tau}{a_\tau}}} \left(1+\frac{\beta^2}{2a_\tau\frac{v_0 b_\tau}{a_\tau}}\right)^{-\frac{2a_\tau+1}{2}}. \qquad (4)
\end{aligned}
$$

The marginal prior for β_{pe} in the peNMIG model has no closed form. The density given in (4) is also the marginal prior $p(\alpha|a_\tau, b_\tau, a_w, b_w, v_0)$ for α in the peNMIG model so that a density transform yields

$$
\begin{aligned}
p(\beta_{pe} = \alpha\xi|a_\tau, b_\tau, a_w, b_w, v_0) &= \\
&= \int p(\alpha|a_\tau, b_\tau, a_w, b_w, v_0) \underbrace{p\left(\frac{\beta_{pe}}{\alpha}\right)}_{=\xi} \frac{1}{|\alpha|} d\alpha \\
&= \int p\left(\frac{\beta_{pe}}{\xi}|a_\tau, b_\tau, a_w, b_w, v_0\right) p(\xi) \frac{1}{|\xi|} d\xi. \qquad (5)
\end{aligned}
$$

Figure 4 shows the two marginal priors for $v_0 = 0.005$, $(a_\tau, b_\tau) = (5, 50)$ and $a_w = b_w$. Values for peNMIG were determined by numerical integration. Note the characteristic shape of the spike-and-slab prior for the marginal prior without parameter expansion: There is a "spike" around zero which corresponds to the contribution of the t-distribution with scale factor $\sqrt{v_0 b_\tau / a_\tau}$ and a "slab" which corresponds to the contribution of the t-distribution with scale factor $\sqrt{b_\tau / a_\tau}$. The prior for peNMIG has heavier tails and an infinite spike at zero (see (6)). It looks similar to the original spike-and-slab prior suggested by Mitchell and Beauchamp [1988], which used a mixture of a point mass in 0 and a uniform distribution on a finite interval, but sampling for our approach has the benefit of conjugate and proper priors.

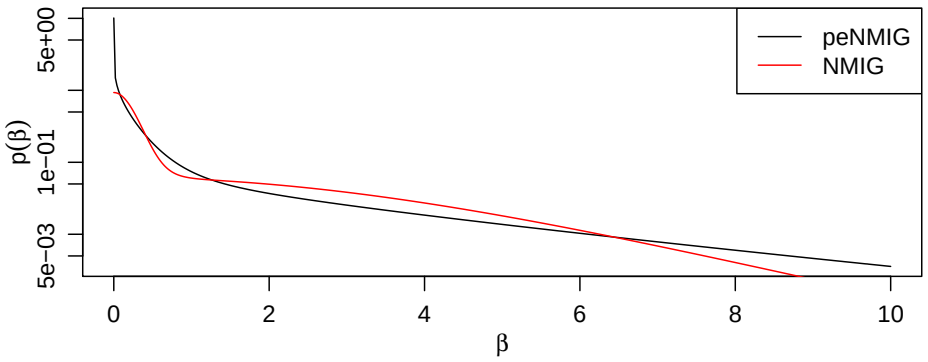

Figure 4: Marginal priors for β as given in (4) and (5) with $(a_\tau, b_\tau) = (5, 50)$, $v_0 = 0.005$, $a_w = b_w$. (Log scale on vertical axis.)

The following shows that the marginal prior $p(\beta_{pe})$ diverges in 0. We use

$$p(\beta_{pe}|a_\tau, b_\tau, a_w, b_w, v_0) = \int_{-\infty}^{+\infty} p_\alpha\left(\frac{\beta_{pe}}{\xi}\right) p_\xi(\xi)\frac{1}{|\xi|}d\xi,$$

so that

$$p(\beta_{pe}|a_\tau, b_\tau, a_w, b_w, v_0)|_{\beta_{pe}=0} = p_\alpha(0)\int_{-\infty}^{+\infty} p_\xi(\xi)\frac{1}{|\xi|}d\xi.$$

It is enough to show that $I = \int_{-\infty}^{+\infty} p_\xi(\xi)\frac{1}{|\xi|}d\xi$ diverges, since $p_\alpha(0)$ is finite and strictly positive. The prior $p_\xi()$ is a mixture of normal densities with variance 1 and means ± 1, so

$$\begin{aligned} I &= K\int_{-\infty}^{+\infty}\frac{1}{|\xi|}\left(\exp\left(-\frac{(\xi+1)^2}{2}\right)+\exp\left(-\frac{(\xi-1)^2}{2}\right)\right)d\xi \\ &= K(I_1+I_2+I_3+I_4) \end{aligned}$$

with

$$I_1 = \int_{-\infty}^{0} -\frac{1}{\xi}\exp\left(-\frac{(\xi+1)^2}{2}\right)d\xi, \qquad I_2 = \int_{0}^{+\infty}\frac{1}{\xi}\exp\left(-\frac{(\xi+1)^2}{2}\right)d\xi,$$
$$I_3 = \int_{-\infty}^{0} -\frac{1}{\xi}\exp\left(-\frac{(\xi-1)^2}{2}\right)d\xi, \text{ and } I_4 = \int_{0}^{+\infty}\frac{1}{\xi}\exp\left(-\frac{(\xi-1)^2}{2}\right)d\xi.$$

Note that $I_1 = I_4$ and $I_2 = I_3$. Since all 4 integrals are positive, it is enough to show that one of them diverges:

$$I_4 = \int_0^1 \underbrace{\frac{1}{\xi}\exp\left(-\frac{(\xi-1)^2}{2}\right)}_{\geq e^{-\frac{1}{2}} \text{ for } \xi\in[0,1]}d\xi + \underbrace{\int_1^{+\infty}\frac{1}{\xi}\exp\left(-\frac{(\xi-1)^2}{2}\right)d\xi}_{=\tilde{K}\geq 0}$$

$$\geq e^{-\frac{1}{2}} \int_0^1 \frac{1}{\xi} d\xi + \tilde{K}$$
$$= e^{-\frac{1}{2}} \left[\ln(\xi)\right]_0^1 + \tilde{K} = +\infty. \tag{6}$$

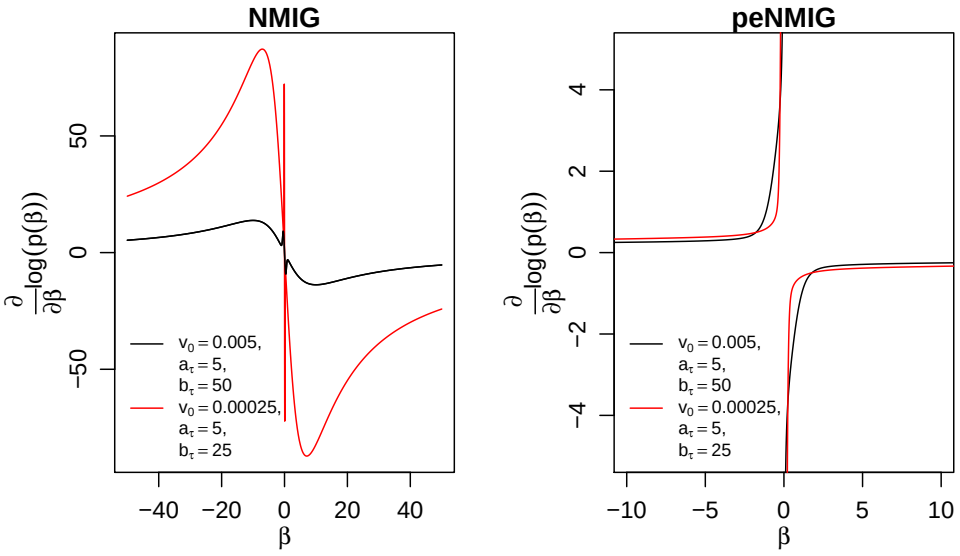

Figure 5: Score functions for marginal priors for beta as given in (4) and (5). Note the different scales for the conventional NMIG and peNMIG.

For both NMIG and peNMIG, the tails of the marginal priors are heavy enough so that they have redescending score functions (see fig. 5) which ensures Bayesian robustness of the resulting estimators. While the shape of peNMIG's score function is similar to that of an $\mathcal{L}_q$-prior with $q \to 0$ and is fairly robust towards different combinations of hyperparameters, the conventional NMIG score function has a complicated shape determined by the interaction of a_τ, b_τ and v_0. Note that the score function of the marginal prior under parameter expansion descends monotonously and much faster.

The marginal prior of the hypervariances for $\beta_{pe} = \alpha\xi$ is given by the density of the product $\gamma\tau^2\xi^2$ since $\beta_{pe}|\gamma, \tau^2, \xi \sim N(0, \gamma\tau^2\xi^2)$. This marginal prior, which is the integral over the product of a mixture of scaled inverse gamma distributions with a noncentral χ^2_1 distribution

$$p(\lambda^2 = \gamma\tau^2\xi^2) =$$
$$= \int_0^\infty \left(\frac{a_w}{a_w + b_w} \Gamma^{-1}\left(\frac{\lambda^2}{\xi^2} | a_\tau, b_\tau\right) + \frac{b_w}{a_w + b_w} \Gamma^{-1}\left(\frac{\lambda^2}{\xi^2} | a_\tau, v_0 b_\tau\right) \right)$$
$$\frac{1}{\xi^2} \chi^2_1(\xi^2 | \mu = 1) d\xi^2,$$

$$\Gamma^{-1}(x|a,b) = \frac{a^b}{\Gamma(a)} x^{-(a+1)} \exp\left(-\frac{b}{x}\right),$$

$$\chi^2_1(x|\mu = 1) = \frac{1}{2} \exp\left(-\frac{x+1}{2}\right) x^{-\frac{1}{4}} I_{-\frac{1}{2}}\left(\sqrt{x}\right),$$

($I_\nu(y)$ denotes the modified Bessel function of the first kind) is intractable, so we are unable to verify whether conditions for Theorem 1 in Polson and Scott [2010] apply. Simulation results indicate that the peNMIG prior has similar robustness for large coefficient values and better sparsity recovery as the horseshoe prior (see p. 30), for which the theorem applies.

The peNMIG prior combines an infinite spike at zero with heavy tails. This desirable combination is similar to other shrinkage priors such as the horseshoe prior [Carvalho et al., 2010] and the normal-Jeffreys prior [Bae and Mallick, 2004] for which both robustness for large values of β and very efficient estimation of sparse coefficient vectors have been shown [Carvalho et al., 2010, Polson and Scott, 2010].

Constraint regions

The shapes of the 2-d constraint regions $\log p((\beta_1, \beta_2)') \leq \text{const}$ implied by the NMIG and peNMIG priors provide some further intuition about their shrinkage properties. The contours of the NMIG prior, depicted on the left

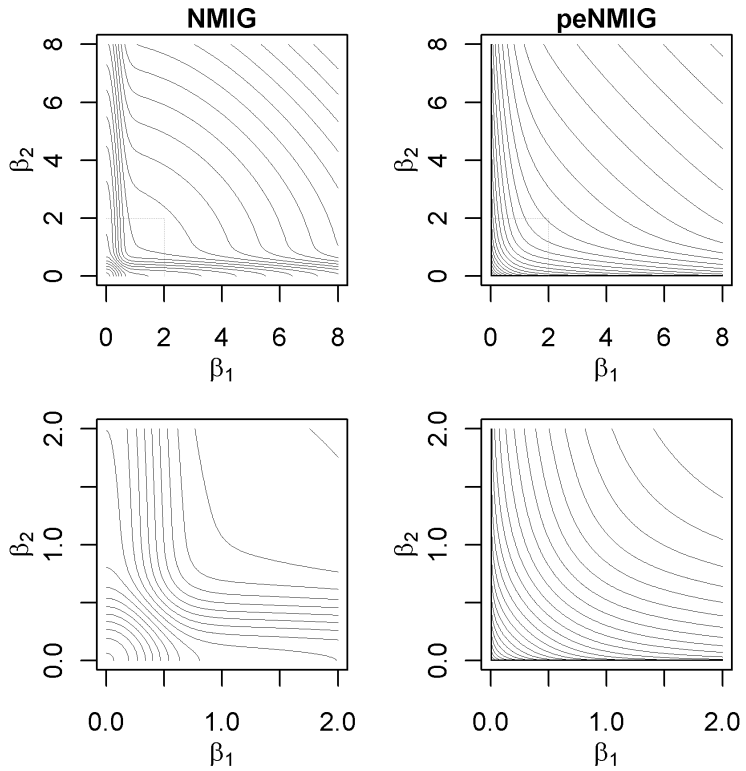

Figure 6: Contour plots of $\log p((\beta_1, \beta_2)')$ for $a_\tau = 5$, $b_\tau = 50$, $v_0 = 0.005$, $a_w = b_w$ for the standard NMIG model and the model with parameter expansion. Lower panels are zooms into the region around the origin (indicated in the upper panels).

in fig. 6, have different shapes depending on the distance from the origin. Close to the origin ($\beta < .3$), they are circular and very closely spaced, im-

plying strong ridge-type shrinkage – coefficient values this small fall into the "spike"-part of the prior and will be strongly shrunk towards zero. Moving away from the origin ($.3 < \beta < .8$), the shape of the contours defining the constraint region morphs into a rhombus shape with rounded corners that is similar to that produced by a Cauchy prior. Still further from the origin ($1 < \beta < 2$), the contours become convex and resemble those of the contours of an $\mathcal{L}_q$ penalty function, i.e. a prior with $p(\beta) \propto \exp(-|\beta|^q)$, with $q < 1$. Coefficient pairs in this region will be shrunk towards one of the axes, depending on their posterior correlation and which of their maximum likelihood estimators is bigger. For even larger β, the shape of the contours is a mixture of a ridge-type circular shape around the bisecting angle with pointy ends close to the axes. The concave shape of the contours in the areas far from the axes implies proportional (i.e. ridge-type) shrinkage of very large coefficient pairs. This corresponds to the comparatively smaller tail robustness of the conventional NMIG prior observed in simulations.

The shape of the constraint region implied by the peNMIG prior has the convex shape of a $\mathcal{L}_q$-penalty function with $q < 1$, which has the desirable properties of simultaneous strong shrinkage of small coefficients and weak shrinkage of large coefficients due to its closeness to the $\mathcal{L}_0$ penalty (see also fig. 8).

Until now, the discussion has been limited to bivariate shrinkage properties applied to single coefficients from *separate* penalization groups. In the following, we discuss shrinkage properties for coefficients from the *same* penalization group, i.e. two entries from the same subvector $\boldsymbol{\beta}_j$ in the notation of Figs. 1 and 3. The shape of the peNMIG prior for 2 coefficients from the same penalization group is quite different. Recall that two coefficients (β_1, β_2) from the same penalization group share the same α, e.g. in this case $(\beta_1, \beta_2)' = \alpha(\xi_1, \xi_2)'$. This results in a very different shape of $\log p((\beta_1, \beta_2)') \leq \text{const}$ shown in Figure 7 (values determined by numerical integration). The prior in this case is

$$
\begin{aligned}
p(\boldsymbol{\beta}_{pe} &= \alpha(\xi_1, \xi_2)' | a_\tau, b_\tau, a_w, b_w, v_0) = \\
&= \int p(\alpha | a_\tau, b_\tau, a_w, b_w, v_0) p\left(\frac{\boldsymbol{\beta}_{pe}}{\alpha}\right) \frac{1}{|\alpha|} d\alpha \\
&= \int p(\alpha | a_\tau, b_\tau, a_w, b_w, v_0) \frac{1}{|\alpha|} \cdot \\
&\quad \cdot \frac{1}{4} \left(N\left(\frac{\beta_1}{\alpha} | \mu = 1\right) + N\left(\frac{\beta_1}{\alpha} | \mu = -1\right)\right) \cdot \\
&\quad \cdot \left(N\left(\frac{\beta_2}{\alpha} | \mu = 1\right) + N\left(\frac{\beta_2}{\alpha} | \mu = -1\right)\right) d\alpha,
\end{aligned}
$$

where $N(x|\mu)$ denotes the normal density with variance 1 and mean μ. The shape of the constraint region for grouped coefficients is that of a square with rounded corners. Compared with the convex shape of the constraint region, this shape induces less shrinkage toward the axes and more towards the origin or along the bisecting angle.

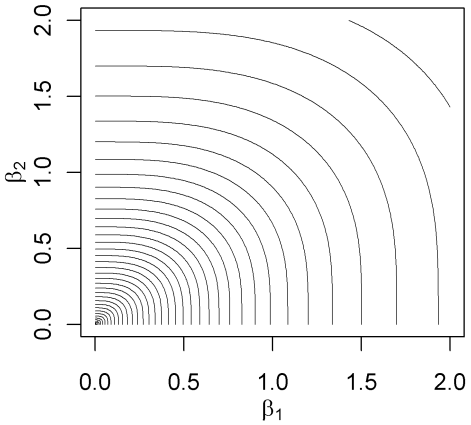

Figure 7: Constraint region for $\boldsymbol{\beta} = (\beta_1, \beta_2)'$ from the same penalization group.

Figure 8 illustrates the difference in shrinkage behavior between grouped and ungrouped coefficients for a simple toy example. We simulated design matrices $\boldsymbol{X}$ with $n = 15$ observations and 2 covariates so that $(\boldsymbol{X'X})^{-1} = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$ with $\rho = -0.8, 0, 0.8$. Coefficients $\boldsymbol{\beta}$ were either $(1, 1)'$ (two intermediate effect sizes) or $(0, 2)'$ (one null, one large effect) and observations y were generated with a signal-to-noise ratio of 2. We generated 100 datasets for each combinations of ρ and $\boldsymbol{\beta}$ and compared OLS estimates to the posterior means for a peNMIG model as returned by spikeSlabGAM.

The different shrinkage properties for grouped and ungrouped coefficients are most apparent for uncorrelated coefficients (middle column): Shrinkage in this case occurs in directions orthogonal to the contours of the prior, so while the shape of the grouped prior causes shrinkage toward the origin in the direction of the bisecting angle or parallel to the axes, the ungrouped coefficients are shrunk more toward the nearest axis. Consequently, we expect estimation error for sparse coefficient vectors with few large and many small or zero entries (like $\boldsymbol{\beta} = (0, 2)'$) to be smaller for ungrouped coefficients, while the grouped prior should have a smaller bias for coefficient vectors with many entries of similar (absolute) size (like $\boldsymbol{\beta} = (1, 1)'$): While most of the mass of the multivariate prior for ungrouped coefficients is concentrated along the axes (i.e. on sparse coefficient vectors), the multivariate prior for grouped coefficients is concentrated in a cube around the origin.

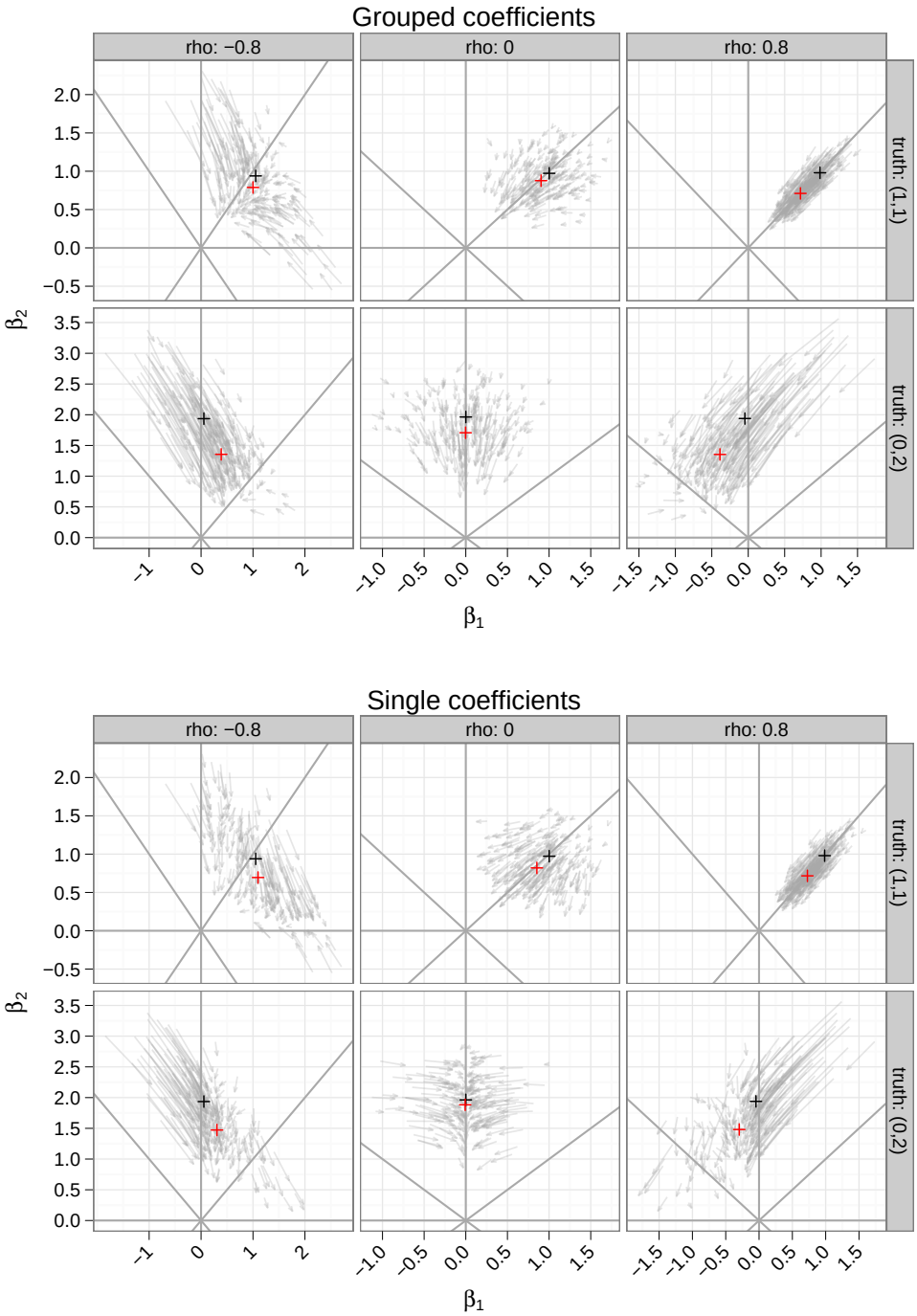

Figure 8: Shrinkage for grouped (top graph) and ungrouped coefficients (bottom graph).
Arrows connect OLS estimates with posterior means from spikeSlabGAM on identical data sets. Black crosses denote means of OLS estimators over all replications for a given setting, red crosses means of posterior means from a peNMIG model fit with spikeSlabGAM. Top rows in each graph are for $\boldsymbol{\beta} = (1,1)'$, bottom rows for $\boldsymbol{\beta} = (0,2)'$. Columns show results for $\rho = -0.8, 0, 0.8$. Note that ρ is the correlation of the OLS estimators, not the correlation of the associated covariates.

4 MCMC

This section describes the MCMC sampler implemented in spikeSlabGAM that was used for all the simulations and applications in Sections 5 and 6. Algorithm 1 on p. 26 gives a short summary of the blockwise Metropolis-within-Gibbs sampler we use.

4.1 Full conditionals

The sampler exploits the fact that the full conditionals of (most of) the parameters are available in closed form:

$$
\begin{aligned}
w|\cdot &\sim \text{Beta}\left(a_w + \sum_j^p \delta_1(\gamma_j), b_w + \sum_j^p \delta_{v0}(\gamma_j)\right), \\
\tau_j^2|\cdot &\sim \Gamma^{-1}\left(a_t + d_j/2, b_t + \frac{\sum_{i=1}^{d_j}\beta_{ji}^2}{2\gamma_j}\right), \\
\frac{P(\gamma_j = 1|\cdot)}{P(\gamma_j = v_0|\cdot)} &= v_0^{d_j/2}\exp\left(\frac{(1-v_0)}{2v_0}\frac{\sum_{i=1}^{d_j}\beta_{ji}^2}{\tau_j^2}\right).
\end{aligned} \tag{7}
$$

Full conditionals for $\boldsymbol{\beta}_j$ for Gaussian responses and the conventional NMIG model (given in fig. 1) are given by

$$
\boldsymbol{\beta}_j|\cdot \sim N(\boldsymbol{\mu}_j, \Sigma_j) \tag{8}
$$

with

$$
\boldsymbol{\Sigma}_j = \left(\frac{1}{\sigma_\varepsilon^2}\boldsymbol{X}_j'\boldsymbol{X}_j + \frac{1}{\gamma_j\tau_j^2}\boldsymbol{I}_{d_j}\right)^{-1};\ \boldsymbol{\mu}_j = \frac{1}{\sigma_\varepsilon^2}\Sigma_j\boldsymbol{X}_j'\boldsymbol{y}.
$$

In the peNMIG model given in fig. 3, updates for $\boldsymbol{\alpha}$ use the "collapsed" design matrix $\boldsymbol{X}_\alpha = \boldsymbol{X}\,\text{blockdiag}(\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_p)$, while $\boldsymbol{\xi}$ is updated based on a "rescaled" design matrix $\boldsymbol{X}_\xi = \boldsymbol{X}\,\text{blockdiag}(\mathbf{1}_{d1}, \ldots, \mathbf{1}_{dp})\boldsymbol{\alpha}$, where $\mathbf{1}_d$ is a $d \times 1$ vector of ones. For Gaussian responses, these are draws from their multivariate normal full conditionals as above. For non-Gaussian responses, we use P-IWLS proposals [Lang and Brezger, 2004] with a Metropolis-Hastings step. The following Section 4.2 provides more details on the methods used to sample $\boldsymbol{\beta}$.

Note that

$$
\frac{P(\gamma_j = 1|\cdot)}{P(\gamma_j = v_0|\cdot)} > v_0^{d_j/2} \text{ for all values of } \boldsymbol{\beta}_j, \text{ i.e that}
$$

$$
P(\gamma_j = 1|\cdot) > \frac{v_0^{d_j/2}}{1 + v_0^{d_j/2}} \approx v_0^{d_j/2} \text{ for small } v_0.
$$

4.2 Updating $\boldsymbol{\beta}_{pe}$

This section describes the implementation of the updates for the regression coefficients in the peNMIG model. For both Gaussian and non-Gaussian responses, the proposed algorithm does blockwise updates of coefficient subvectors, conditional on the remainder of the coefficient vector and the other parameters in the Markov blanket (i.e. prior covariances, prior means and the relevant likelihood terms). The default is a blocksize of 30 for both $\boldsymbol{\alpha}$ and $\boldsymbol{\xi}$ for Gaussian response and smaller blocksizes of 5 and 15 for $\boldsymbol{\alpha}$ and $\boldsymbol{\xi}$, respectively, for non-Gaussian response. Blocksizes are smaller for non-Gaussian response since the acceptance probability in the necessary Metropolis-Hastings-step for non-Gaussian responses tends to decrease quickly with increasing dimension of the proposal.

Since $\boldsymbol{\beta} = \text{blockdiag}(\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_p)\boldsymbol{\alpha}$, we sample $\boldsymbol{\beta}$ by first updating $\boldsymbol{\alpha}$ based on a "collapsed" $n \times p$ design matrix $\boldsymbol{X}_\alpha = \boldsymbol{X}\,\text{blockdiag}(\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_p)$ and then updating $\boldsymbol{\xi}$ based on a "rescaled" $n \times q$ design matrix $\boldsymbol{X}_\xi = \boldsymbol{X}\,\text{blockdiag}(\mathbf{1}_{d1}, \ldots, \mathbf{1}_{dp})\boldsymbol{\alpha}$, where $\mathbf{1}_d$ is a $d \times 1$ vector of ones. The j-th column of $\boldsymbol{X}_\alpha$ contains the sum of the original design columns multiplied by the entries in the subvector $\boldsymbol{\xi}_j$ associated with α_j. Each column in $\boldsymbol{X}_\xi$ contains the respective column of the original design matrix multiplied by the associated entry in $\boldsymbol{\alpha}$. The prior means $m_l \in \{\pm 1\}$ for $\xi_l \sim N(m_l, 1)$ are drawn beforehand from their full conditionals via $P(m_l = 1|\cdot) = \frac{1}{1+\exp(-2\xi_l)}$.

Update via QR-decomposition

The following paragraphs describe a general method to update a coefficient vector $\boldsymbol{\delta}$ associated with a conditional Gaussian prior. We use this procedure to update $\boldsymbol{\beta}$ in the NMIG model and to update both $\boldsymbol{\alpha}$ and $\boldsymbol{\xi}$ in the peNMIG model.

Regression coefficients $\boldsymbol{\delta}$ with prior $\boldsymbol{\delta} \sim N(\boldsymbol{\mu}^\delta, \boldsymbol{\Sigma}^\delta)$ and associated design matrix $\boldsymbol{X}^\delta$ can be updated by running the regression of an augmented data vector $\tilde{\boldsymbol{y}}$ with covariance $\widetilde{\boldsymbol{\Sigma}}$ on an augmented design matrix $\tilde{\boldsymbol{X}}$ with

$$\tilde{\boldsymbol{y}} = \begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{\mu}^\delta \end{pmatrix}; \quad \tilde{\boldsymbol{X}} = \begin{pmatrix} \boldsymbol{X}^\delta \\ \boldsymbol{I} \end{pmatrix} \text{ and } \widetilde{\boldsymbol{\Sigma}} = \begin{pmatrix} \text{Cov}(\boldsymbol{y}) & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}^\delta \end{pmatrix}. \tag{9}$$

If only a subvector $\boldsymbol{\delta}_j$ is updated conditional on the remainder $\boldsymbol{\delta}_{-j}$ of the vector $\boldsymbol{\delta}$, $\boldsymbol{y}$ is replaced by $\boldsymbol{y} - \boldsymbol{X}^\delta_{-j}\boldsymbol{\delta}_{-j}$ and $\boldsymbol{\Sigma}^\delta$ is replaced by $\boldsymbol{\Sigma}^\delta_{-j,-j}$.

Following Gelman et al. [2008], we perform the updates for the regression coefficients via the QR-decomposition $\widetilde{\boldsymbol{\Sigma}}^{-1/2}\tilde{\boldsymbol{X}} = \boldsymbol{QR}$. From this decomposition, we can solve the triangular system $\boldsymbol{R}\hat{\boldsymbol{\delta}} = \boldsymbol{Q}\left(\widetilde{\boldsymbol{\Sigma}}^{-1/2}\tilde{\boldsymbol{y}}\right)$ for the mean of the full conditional $\hat{\boldsymbol{\delta}}$. As long as $\widetilde{\boldsymbol{\Sigma}}^{-1/2}$ is a diagonal matrix, as is the case for all of the models and predictor terms we are considering (see Section 2.3), or is known, the computationally demanding step is the computation of the QR-decomposition.

We solve another triangular system $\boldsymbol{R}\boldsymbol{e}_\delta = \boldsymbol{n}$, $n_i \overset{\text{i.i.d.}}{\sim} N(0,1)$ in order to generate a candidate value $\delta^c = \hat{\boldsymbol{\delta}} + \boldsymbol{e}_\delta$ from (the approximation to) the full conditional, so the proposal distribution $q(\delta^c, \delta)$ is $N(\hat{\boldsymbol{\delta}}, (\boldsymbol{R}'\boldsymbol{R})^{-1})$.

IWLS updates for non-Gaussian responses

We use a variant of the well-known IWLS proposal scheme [Gamerman, 1997] to do blockwise updates for both $\boldsymbol{\alpha}$ and $\boldsymbol{\xi}$ in the non-Gaussian case. We use a penalized IWLS (P-IWLS) proposal scheme based on an approximation of the current posterior mode described in detail in Brezger and Lang [2006] (Sampling scheme 1, Section 3.1.1). This method is a Metropolis-Hastings type update which uses a Gaussian (i.e. second order Taylor) approximation to the full conditional around its approximate mode as its proposal distribution. The approximating Gaussian is obtained by performing a single Fisher scoring step per iteration.

For P-IWLS, $\boldsymbol{y}$ and $\text{Cov}(\boldsymbol{y})$ in (9) are replaced by their IWLS equivalents [Gamerman, 1997]

$$\text{Cov}(\boldsymbol{y}) \overset{IWLS}{\approx} \text{diag}\left(b''(\boldsymbol{\theta})g'(\boldsymbol{\mu})^2\right) \text{ and } \boldsymbol{y} \overset{IWLS}{\approx} \boldsymbol{X}_j\boldsymbol{\delta}_j + (\boldsymbol{y} - \boldsymbol{\mu})g'(\boldsymbol{\mu}), \tag{10}$$

see (1) for notation.

We use the following modification of the IWLS-algorithm in order to decrease the computational complexity of the algorithm somewhat: By using the mean of the proposal distribution of the previous iteration $\hat{\boldsymbol{\delta}}^p$ instead of $\boldsymbol{\delta}$ in (10) and recalculating $\boldsymbol{\mu}$ and $\boldsymbol{\theta}$ based on $\hat{\boldsymbol{\delta}}^p$, the proposal distribution $q()$ becomes independent of the current state, which simplifies the calculation of the acceptance probability and can increase acceptance rates [Brezger and Lang, 2006].

Acceptance rates for the sampler strongly depend on the size of the update blocks and on the magnitude of the rescaling performed in each iteration: For large blocks or updates that require drastic rescaling (see paragraph below), acceptance probabilities can occasionally become small, especially for binary responses. To avoid getting stuck, our sampler monitors rejection rates for each block. If proposals for a certain update block have been rejected 10 times in a row, we use a different proposal density for this block with probability 0.5: Instead of drawing proposals from $N(\hat{\boldsymbol{\delta}}^p, (\boldsymbol{R}'\boldsymbol{R})^{-1})$, we use $q(\delta^c, \delta) = N(\delta^c, (\boldsymbol{R}'\boldsymbol{R})^{-1})$, i.e. we use the current state as the mean of the proposal. The working observations and IWLS weights that determine $\boldsymbol{R}$ are calculated from the mode of the previous iteration as described above so that the proposal ratio $q(\delta, \delta^c)/q(\delta^c, \delta)$ is 1. This type of update tends to result in smaller steps, but it is useful in order to get the chain moving again. Using an adaptive transition kernel such as this one can violate the detailed balance condition for the transition kernel of the Markov chain, but results in Section 5 convincingly show that convergence of the chains is not adversely affected. For most datasets, mode switching occurs very rarely during the sampling of the chain if at all, and spikeSlabGAM provides the option to switch it off

entirely. Direct comparisons of results on problematic datasets between exceedingly long (i.e. > 100000 iterations for a model with 20 coefficients) runs of single-site-IWLS-updates without mode switching and blocked updates with mode switching showed that differences between the resulting posterior distributions were well within the range of MC error.

Rescaling parameter blocks

After updating the entire $\boldsymbol{\alpha}-$ and $\boldsymbol{\xi}-$vectors, each subvector $\boldsymbol{\xi}_j$ is rescaled so that $|\boldsymbol{\xi}_j|$ has mean 1, and the associated α_j is rescaled accordingly so that $\boldsymbol{\beta}_j = \alpha_j \boldsymbol{\xi}_j$ is unchanged:

$$\boldsymbol{\xi}_j \to \frac{d_j}{\sum_i^{d_j} |\xi_{ji}|} \boldsymbol{\xi}_j \quad \text{and} \quad \alpha_j \to \frac{\sum_i^{d_j} |\xi_{ji}|}{d_j} \alpha_j.$$

This rescaling is advantageous since α_j and $\boldsymbol{\xi}_j$ are not identifiable and thus their sampling paths can wander off into extreme regions of the parameter space without affecting the fit, e.g. α_j becoming extremely large while entries in $\boldsymbol{\xi}_j$ simultaneously become extremely small. By rescaling, we retain the interpretation of α_j as a scaling factor representing the importance of the model term associated with it and avoid numerical problems that can occur for extreme parameter values. For non-Gaussian responses, the posterior modes used in the IWLS-updates are shifted accordingly as well. Note, however, that this shifting of the mode is only approximate. Consequentially, this rescaling can occasionally lead to low ($< .1$) acceptance rates for the P-IWLS proposals since the proposal density may not be well adapted to the posterior anymore after a large rescaling.

Starting values

It is essential to find suitable starting values for $\boldsymbol{\beta}$ for non-Gaussian responses, otherwise the IWLS sampler fails. We initialize $\boldsymbol{\beta}$ by performing Fisher scoring steps with fixed and usually large values of the hypervariance until the relative change in $\boldsymbol{\beta}$ are smaller than 10%, up to a maximum of 20 steps. Starting values for $\boldsymbol{\alpha}^{(0)}$ and $\boldsymbol{\xi}^{(0)}$ are computed via

$$\alpha_j^{(0)} = \frac{\sum_i^{d_j} |\beta_{ji}|}{d_j} \quad \text{and} \quad \boldsymbol{\xi}_j^{(0)} = \frac{\boldsymbol{\beta}_j}{\alpha_j^{(0)}}.$$

Simulation results and applications (Sections 5, 6) show that this strategy works well.

4.3 Estimating Inclusion Probabilities

Selection of coefficient blocks $\boldsymbol{\beta}_j$ in the NMIG and peNMIG models is based on the marginal posterior of γ_j. The posterior expectation of $\delta_1(\gamma_j)$ is the

Algorithm 1 MCMC sampler for peNMIG

Initialize $\boldsymbol{\tau}^{2(0)}, \boldsymbol{\gamma}^{(0)}, \sigma^{2(0)}, w^{(0)}$ and $\boldsymbol{\beta}^{(0)}$ (via IWLS for non-Gaussian response as described on p. 25)
Compute $\boldsymbol{\alpha}^{(0)}, \boldsymbol{\xi}^{(0)}, \boldsymbol{X}_{\alpha}^{(0)}$
for iterations $t = 1, \ldots, T$ **do**
  for blocks $b = 1, \ldots, B_{\alpha}$ **do**
    generate $\boldsymbol{\alpha}_b^{(t)}$ from its full conditional (Gaussian case)/ via IWLS-P
  $\boldsymbol{X}_{\xi}^{(t)} = \boldsymbol{X}\,\text{blockdiag}(\mathbf{1}_{d_1}, \ldots, \mathbf{1}_{d_p})\boldsymbol{\alpha}^{(t)}$
  generate $m_1^{(t)}, ..., m_q^{(t)}$ from their full conditionals
  for blocks $b = 1, \ldots, B_{\xi}$ **do**
    generate $\boldsymbol{\xi}_b^{(t)}$ from its full conditional (Gaussian case)/ via IWLS-P
  for penalization groups $i = 1, \ldots, p$ **do**
    rescale $\boldsymbol{\xi}_i^{(t)}$ and $\alpha_i^{(t)}$ (see p. 25)
  $\boldsymbol{X}_{\alpha}^{(t)} = \boldsymbol{X}\,\text{blockdiag}(\boldsymbol{\xi}_1^{(t)}, \ldots, \boldsymbol{\xi}_p^{(t)})$
  generate ${\tau_1}^{2(t)}, ..., {\tau_p}^{2(t)}$ from their full conditionals
  generate ${\gamma_1}^{(t)}, ..., {\gamma_p}^{2(t)}$ from their full conditionals
  generate $w^{(t)}$ from its full conditional
  if $\boldsymbol{y}$ is Gaussian **then**
    generate $\sigma^{2(t)}$ from its full conditional

posterior inclusion probability $p_{in,j}$, since $p_{in,j} = P(\gamma_j = 1) = E(\delta_1(\gamma_j))$. Inclusion probabilities $p_{in,j}$ are estimated with the Rao-Blackwellized estimator

$$\widehat{p_{in,j}} = T^{-1} \sum_{t=0}^{T} p_{in,j}^{(t)},$$

$$\text{with } p_{in,j}^{(t)} = 1 - \begin{cases} \left(1 + v_0^{d_j/2} \exp\left(\frac{(1-v_0)}{2v_0} \frac{\sum_{i=1}^{d_j} (\beta_{ji}^{(t)})^2}{(\tau_j^2)^{(t)}}\right)\right)^{-1} & \text{for NMIG,} \\ \left(1 + v_0^{1/2} \exp\left(\frac{(1-v_0)}{2v_0} \frac{(\alpha_j^{(t)})^2}{(\tau_j^2)^{(t)}}\right)\right)^{-1} & \text{for peNMIG,} \end{cases}$$

where $\theta^{(t)}$ denotes the realized value of parameter θ in iteration t of an MCMC chain with length T. This estimator uses the MCMC samples of $P(\gamma_j = 1)$ after burn-in, instead of $\widehat{p_{in,j}} = T^{-1} \sum_{t=0}^{T} \delta_1(\gamma_j^{(t)})$.

Barbieri and Berger [2004] show that, under fairly strong conditions, the *median probability model*, i.e. the model which includes only covariates with a marginal inclusion probability greater than 0.5, is optimal for predictive purposes in the class of single models. Although the conditions set forth (i.e. orthogonal design, squared error loss) do not apply to most of the settings in which spikeSlabGAM could conceivably be used, we still use this threshold of $p_{in,j} = 0.5$ to determine exclusion or inclusion of model terms in the following. We concur with their assertion that "[...] the fact that *only* the median probability model seems to have any optimality theory whatsoever suggests

that it might quite generally be successful, even when the optimality theory does not apply" [Barbieri and Berger, 2004, p. 894] and this is borne out by simulation studies and applications (see Sections 5, 6).

4.4 Algorithm Variants

While the default prior for the inclusion indicators γ_j assumes mutual independence, i.e. that inclusion or exclusion of a model term is a priori independent of the inclusion or exclusion of all other model terms, we also implemented a structure of the prior for $\boldsymbol{\gamma}$ that incorporates the hierarchical structure of the model terms themselves. More precisely, the prior structure forces inclusion of e.g. the linear term for a covariate if the corresponding smooth term is included in the model, or the inclusion of main effects if an interaction effect involving them is included in the model. Without changing the sampler per se, this "top-down" approach is implemented as a simple pass over the updated $\boldsymbol{\gamma}$-vector in each iteration, making sure that all low-order terms (i.e. main effects) have $\gamma = 1$ if high-order terms that involve them (i.e. interactions) have $\gamma = 1$. Alternatively, a "bottom-up" variant enforcing more parsimonious models that excludes high-order terms (i.e. sets them to $\gamma = v_0$) unless all low-order terms associated with them are included may be an option worth pursuing, but we have not done so yet. An alternative to be implemented in future versions of the software is to sample $\boldsymbol{\gamma}$ not via single-site updates, but blockwise with blocks determined by the dependencies induced by the hierarchy (e.g. sample γs for main effects and their interaction together) and then include a Metropolis-Hastings step to reject proposals that violate the hierarchical constraints in a block.

5 Simulation Studies

The following sections summarize results from tests of the proposed methods on simulated data. Section 5.1 investigates the adaptive shrinkage properties of the proposed prior. Section 5.2 shows that the proposed parameter expansion with multiplicative redundant parameters can improve sampling behavior for settings in which the posterior of the regression coefficients contains strong correlations. Sections 5.3 and 5.4 investigate model selection and estimation performance for models with random intercepts and smooth functions, respectively. Section 5.5 describes results for additive models of some complexity for both Gaussian and Poisson responses and compares the performance of our approach to the performances of other recently suggested algorithms.

We introduce some additional notation for the generation of Gaussian data: For a given data-generating process (DGP) that generates a random design matrix $\boldsymbol{X}$ and a (fixed or random) vector of coefficients $\boldsymbol{\beta}$, let $\boldsymbol{\eta} = \boldsymbol{X\beta}$ denote the "true" linear predictor. For responses with $\boldsymbol{y} = \boldsymbol{\eta} + \boldsymbol{\varepsilon}$, the difficulty level of estimating both $\boldsymbol{\beta}$, and, consequently, $\boldsymbol{\eta}$ is determined mostly by the ratio between the systematic variability that can be quantified as the observed

variability of $\boldsymbol{\eta}$, i.e. the "signal", and the unsystematic variability introduced by the Gaussian error terms $\boldsymbol{\epsilon}$, the "noise". Let $\text{sd}_\eta = \sqrt{\sum_i^n (\eta_i - \bar{\eta})^2 / n}$ and define the signal-to-noise ratio $\text{SNR} = n\,\text{sd}_\eta^2 \,/\, \sum_i^n \varepsilon_i^2$. For a given value of SNR and realization of $\boldsymbol{\eta}$, responses $\boldsymbol{y}$ are then generated via $y_i \sim N\left(\eta_i, \text{sd}_\eta^2 \,/\, \text{SNR}\right)$.

5.1 Adaptive shrinkage

We investigate the shrinkage properties of the proposed prior structures in a simple setting. The following describes the data-generating process:

- $n = 20, 50, 100$ observations
- $\beta = (.1, .2, .3, \ldots, 1), p = 10$
- signal-to-noise ratio SNR$= 0.5, 2$
- covariates $\boldsymbol{x}_j$ are independent, with $\boldsymbol{x}_j \sim U[-2, 2]$ and enter the model scaled to have mean 0 and standard deviation .5.
- 100 replications per setting

We compare the shrinkage properties of the posterior means from spikeSlabGAM with those of the horseshoe prior (HS) as implemented in R package monomvn [Gramacy, 2010] and the LASSO estimator (L1) as implemented in R package lasso2 [Lokhorst et al., 2009]. The horseshoe prior (a scale mixture of normals with a scaled half-Cauchy mixing distribution, where the scale of the mixing distribution is itself half-Cauchy distributed), has recently been shown to have excellent adaptive shrinkage properties [Carvalho et al., 2010] and we use its behavior as a reference for good adaptive shrinkage properties, while the LASSO estimators serve as a reference for a shrinkage estimator without adaptivity.

Figure 9 shows the median and the inter-quartile ranges of the posterior means of the estimated coefficients over the 100 replications for each combination of the different numbers of observations n and the signal-to-noise ratios SNR. We compare models with (peNMIG) and without (NMIG) the redundant multiplicative parameter expansion with $(a_\tau, b_\tau, v_0) = (5, 25, 0.00025)$ or $(5, 50, 0.005)$.

Note that the frequentist LASSO (L1, in yellow) performs about the same amount of regularization in all of the settings – all six approaches overshrink the larger coefficients for $N = 20$ and $N = 50$, SNR$= 0.5$; LASSO less so than the Bayesian approaches. However, as more information from the data becomes available with increasing N and SNR, the Bayesian approaches (NMIG, peNMIG, HS) perform less regularization, since the likelihood contribution of the posterior increasingly dominates the prior contribution to the posterior. This is visible especially for the bottom right panel with $N = 100$ and SNR$= 2$.

Adaptive shrinkage in the sense of strong regularization of smaller coefficients (i.e. $\beta \leq 0.5$) and simultaneously weak shrinkage for large coefficients

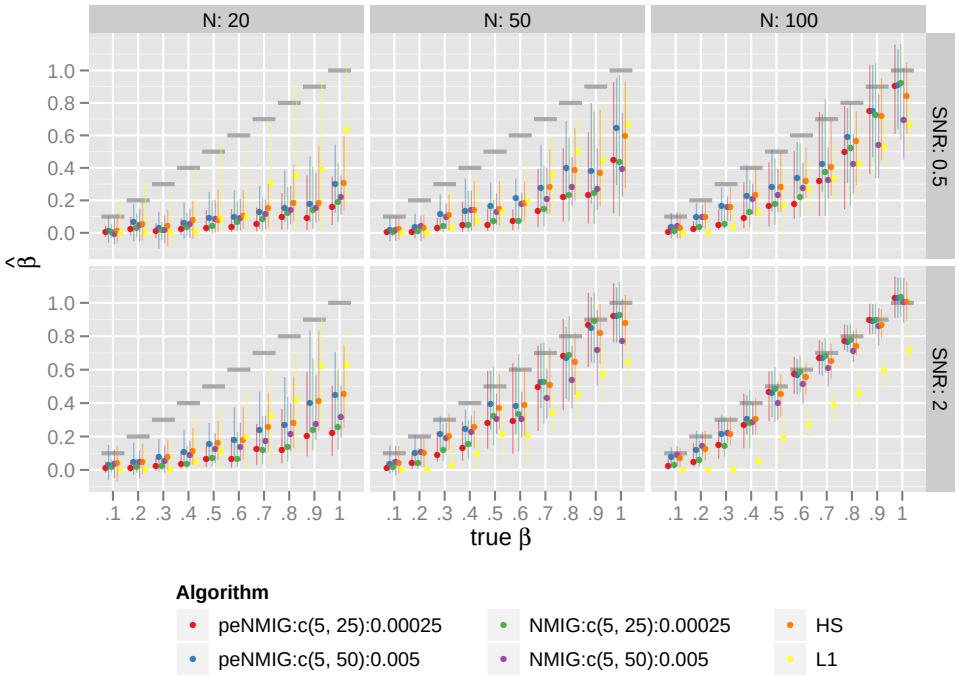

Figure 9: Estimated coefficients (median & inter-quartile range) for different (pe)NMIG-prior settings, the horseshoe prior (HS) and the frequentist LASSO (L1). Fat dark gray horizontal bars show values of the true coefficients.

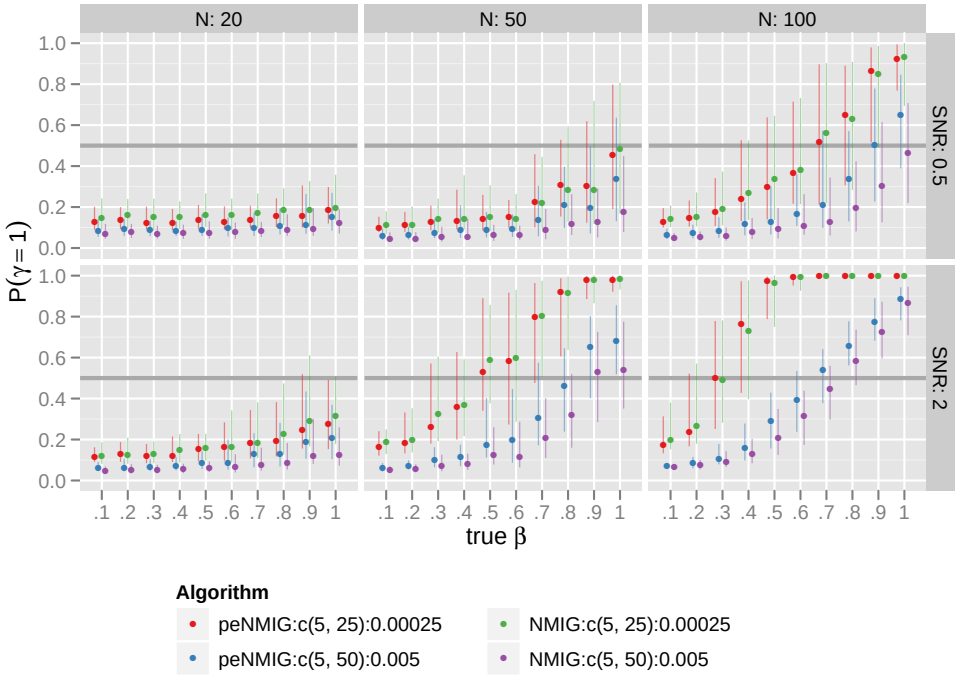

Figure 10: Posterior means of $P(\gamma = 1)$ (median & inter-quartile range) for different NMIG-prior settings.

(i.e. $\beta \geq 0.8$) is observable only for $N = 50, 100$. For $N = 20$, posterior means for peNMIG with $(a_\tau, b_\tau) = (5, 50)$ and $v_0 = 0.005$ are closest to those returned by the horseshoe-prior model. We observe no systematic differences between the shrinkage properties of NMIG and peNMIG for $v_0 = .005$. Estimates and inclusion probabilities (see fig. 10) for the larger coefficients are much smaller for the NMIG model. We also note that inclusion probabilities for peNMIG seem to be somewhat less sensitive to the different hyperparameters than for NMIG. Shrinkage of the smaller coefficients is more pronounced for smaller v_0 and τ^2 (red and green symbols) without a corresponding increase in estimation bias for the larger coefficients, at least for settings with enough data (i.e. $n = 50$, SNR$= 2$ and $n = 100$). For settings with $n = 50$, SNR$= 2$ or $n = 100$, larger v_0 and τ^2 NMIG models without parameter expansion (in purple) perform much worse. This is due to lower inclusion probabilities (see fig. 10). In general, we find that the spikeSlabGAM estimates are similar to the HS estimates.

Across all settings, estimation times for spikeSlabGAM for both NMIG and peNMIG were about one third to half of those for monomvn. In absolute terms, running 3000 iterations of the chains took between 0.16 and 0.36 seconds for spikeSlabGAM depending on n and whether parameter expansion was used, while monomvn's horseshoe implementation took between 0.58 and 0.64 seconds on a modern desktop PC (Intel Core2 Quad Q9550 CPU with 2.83GHz).

Tail robustness and sparsity recovery

In order to compare the robustness of our approaches to large coefficient values relative to that of the horseshoe prior, we replicate the simulation study in Section 3.1. of Polson and Scott [2010]. We simulate 100 datasets with $n = 60$ observations and $p = 40$ covariates. The covariates are independent standard normal variates. The true coefficient vector is 80% sparse, with the first 32 entries equal to zero (i.e. the "noise" component) and the remaining 8 drawn from a t-distribution with 3 degrees of freedom (i.e. the "signal" component). We simulate responses $\boldsymbol{y}$ with normal errors so that the signal-to-noise ratio is 2. Results are shown for prior settings $a_\tau = 5, b_\tau = 50, v_0 = 0.00025, a_w = b_w = 1$ and the default settings for the horseshoe prior as implemented in monomvn. Figure 11 shows the mean square estimation errors (MSE) for posterior means of $\boldsymbol{\beta}$ separately for the noise (upper panel) and signal (lower panel) components of $\boldsymbol{\beta}$. MSE for the noise part is consistently higher for the horseshoe estimates (average MSE-ratio is 2.8 compared to the spikeSlabGAM-estimates for peNMIG and 5.0 for NMIG), while the MSE for the signal part is slightly lower (average MSE-ratio: 0.94 for peNMIG and 0.83 for NMIG). These results show satisfactory tail robustness for both approaches comparable to that of the horseshoe prior and excellent sparsity recovery. As expected (see Section 3.4, figs. 4, 6), robustness is stronger for peNMIG than for NMIG. Sparsity recovery is very good for both of our approaches. We observed qualitatively similar results for signal-to-noise ratios 5 and .5 (not shown).

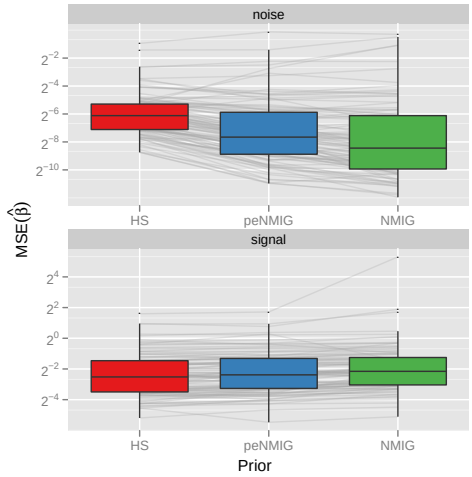

Figure 11: Mean square estimation errors (MSE) for posterior means of $\boldsymbol{\beta}$. Upper panel shows MSE($\hat{\boldsymbol{\beta}}$) for the coefficients that are zero, lower panel shows MSE($\hat{\boldsymbol{\beta}}$) for the coefficients drawn from t_3. Dark grey lines connect values from the same replicates.

5.2 Sampling performance with parameter expansion

We investigate the approximate integrated autocorrelation times – defined as

$$\text{IAT}(\boldsymbol{x}) = \frac{1}{2} + \sum_{t=1}^{T} \hat{r}(t),$$

$\hat{r}(t)$ are the estimated auto correlations for lag t [Jackman, 2009]– for the regression coefficients and their estimation error in designs with strong correlations in the posterior distribution of $\boldsymbol{\beta}$. We generate random design matrices $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ so that $\Psi = (\boldsymbol{X}'\boldsymbol{X})^{-1}$ is a matrix with 1 on the diagonal and a constant ρ everywhere else, i.e. the correlations between all the OLS-estimators are equal to ρ. Specifically, $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Psi}^{-1/2}$, where $\boldsymbol{U}$ is an orthonormal matrix and $\boldsymbol{\Psi}^{-1/2}$ is the Cholesky root of $\boldsymbol{\Psi}^{-1}$. Responses $\boldsymbol{y}$ are then generated as

$$\boldsymbol{y} \sim N_n\left(\boldsymbol{\eta}, \frac{\text{sd}_\eta^2}{\text{SNR}}\boldsymbol{I}_n\right).$$

Regression coefficients $\boldsymbol{\beta}$ are set as an equidistant descending sequence of length 10 from 2 to .5 interspersed with zeroes, i.e. $\boldsymbol{\beta} = (2, 0, 1.8\bar{3}, \ldots, 0.\bar{6}, 0, 0.5, 0)'$ so that $p = 20$.

We use the following settings for our simulations:

- correlation of $\boldsymbol{\beta}_{OLS}$: $\rho = .9, .95$
- signal-to-noise-ratio SNR $= 1, 3$
- no of observations: $n = 50, 100$

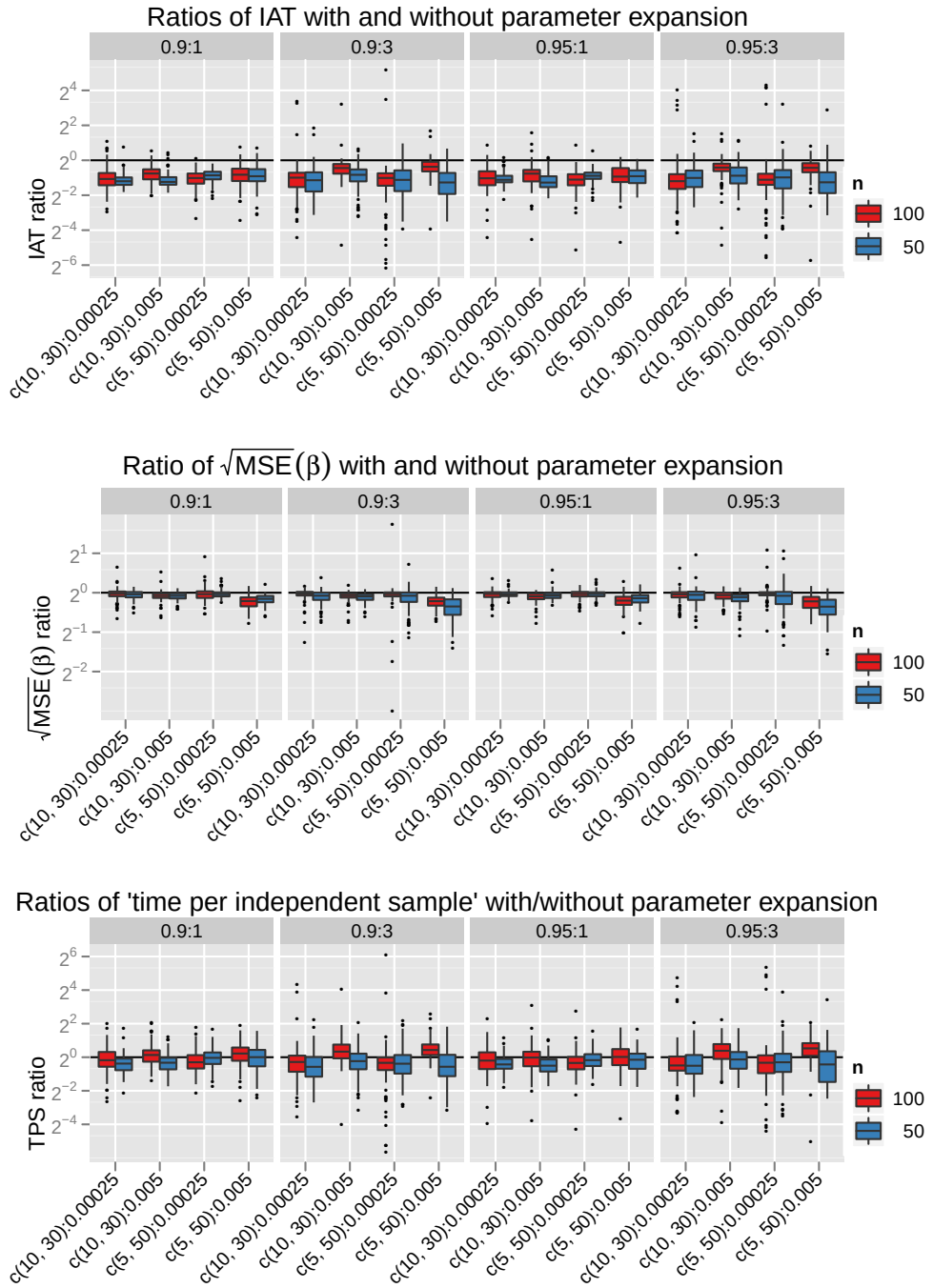

Figure 12: Ratios of average integrated autocorrelation times for $\hat{\boldsymbol{\beta}}$ (upper graph), root mean square estimation error $\sqrt{\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|^2}$ (middle graph) and time per "independent" sample (bottom graph).
Columns correspond to the settings of the data generating process (correlation and SNR). Boxplots contain the ratio between peNMIG and NMIG results for each replicate. Boxplots are grouped into the four different prior settings. Red boxplots correspond to results for $n = 100$ observations, blue for $n = 50$. Vertical axes are on binary log scale; fat black horizontal line corresponds to a ratio of 1, i.e no change.

- 100 replications for each setting

Figure 12 shows the ratios between average integrated autocorrelation times for $\hat{\boldsymbol{\beta}}$ (top graph) and root mean square estimation error $\sqrt{\|\hat{\beta} - \beta\|_2^2}$ (middle graph) with and without parameter expansion for the different settings for the posterior correlation, the signal-to-noise ratio and hyperparameters (a_τ, b_τ) and v_0. Panels from left to right show results for correlation 0.9 with SNR 1 and SNR 3 followed by results for correlation 0.95 with SNR 1 and SNR 3. The simulation shows that the suggested parameter expansion improves mixing and reduces estimation error for all DGP settings and hyperparameter configurations, especially for higher SNR, smaller v_0, and low number of observations. Parameter expansion reduces estimated integrated autocorrelation times for $\boldsymbol{\beta}$ by a median factor of .49 for $n = 50$ and .57 for $n = 100$ and estimation error $\sqrt{\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|_2^2}$ by a median factor of .94 for $n = 50$ and .95 for $n = 100$. Because of the larger complexity of the sampler for peNMIG (see Section 4), the observed improvement in mixing is not large enough to translate into consistent reductions in computing time for $n = 100$: The bottom graph in Figure 12 shows that the time to generate a single "independent sample" (defined as the total run time of the sampler divided by the effective sample size, which is in turn the number of iterations of the chain divided by 2IAT [Jackman, 2009]) remains about the same in most settings, with median ratios of estimated time per independent sample of .80 for $n = 50$ and .98 for $n = 100$. Regression analyses of the simulation results with performance measures as dependent variables and second-degree interactions and main effects for the data-generating process (n, SNR, correlation) and the hyperparameters $((a_\tau, b_\tau), v_0)$ also show that using peNMIG increases the odds of correctly including a covariate in the model by a factor of 1.11, without a corresponding decrease in specificity. Accuracy increases by a factor of 1.04. Table 1 gives mean performance measures for the different settings and priors.

In summary, these results indicate that parameter expansion has the potential to improve mixing for difficult data situations dramatically, although this may not translate into relevant savings in computation time for larger data sets with many parameters due to the higher computational burden of sampling from the parameter expanded posterior. Parameter expansion also reduces estimation error and improves complexity recovery. Note that we did not investigate whether these advantages disappear if the sampler for the conventional NMIG model is allowed to run long enough to achieve a similar effective sample size as that of the parameter expanded model.

5.3 Random Intercept Models

This section summarizes simulation results on selecting and estimating random intercept coefficients for Gaussian and binomial response. The basic data generating process for both types of response is

$$\boldsymbol{\eta} = \boldsymbol{x} + \boldsymbol{Zb}$$

DGP	Prior	Parameter Expansion	Sensitivity	Specificity	IAT	$\sqrt{\text{MSE}(\hat{\beta})}$	TPS $[ms]$
0.9:100:1	c(10, 30):0.00025	Yes	0.69	0.84	4.21	0.09	7.96
		No	0.70	0.80	8.18	0.09	8.50
	c(10, 30):0.005	Yes	0.68	0.86	1.79	0.09	3.36
		No	0.65	0.85	2.97	0.09	2.83
	c(5, 50):0.00025	Yes	0.60	0.93	4.36	0.10	8.22
		No	0.55	0.94	8.44	0.10	8.74
	c(5, 50):0.005	Yes	0.51	0.96	1.14	0.09	2.02
		No	0.37	0.98	2.18	0.11	1.85
0.9:100:3	c(10, 30):0.00025	Yes	0.93	0.95	3.40	0.04	6.31
		No	0.92	0.93	5.83	0.04	6.50
	c(10, 30):0.005	Yes	0.91	0.96	0.93	0.04	1.78
		No	0.89	0.97	1.53	0.04	1.58
	c(5, 50):0.00025	Yes	0.91	0.98	1.90	0.04	3.49
		No	0.89	0.97	4.74	0.04	5.50
	c(5, 50):0.005	Yes	0.77	1.00	0.77	0.04	1.40
		No	0.69	1.00	1.08	0.05	1.04
0.9:50:1	c(10, 30):0.00025	Yes	0.38	0.86	3.86	0.14	5.28
		No	0.34	0.85	8.62	0.15	6.76
	c(10, 30):0.005	Yes	0.38	0.87	1.68	0.14	2.29
		No	0.29	0.88	3.76	0.15	2.90
	c(5, 50):0.00025	Yes	0.19	0.97	4.40	0.16	6.04
		No	0.17	0.97	8.10	0.16	6.29
	c(5, 50):0.005	Yes	0.24	0.96	1.20	0.14	1.59
		No	0.13	0.99	2.38	0.16	1.79
0.9:50:3	c(10, 30):0.00025	Yes	0.77	0.89	4.11	0.07	5.84
		No	0.76	0.85	8.95	0.08	7.79
	c(10, 30):0.005	Yes	0.76	0.91	1.63	0.07	2.06
		No	0.72	0.90	3.20	0.08	2.74
	c(5, 50):0.00025	Yes	0.68	0.96	4.41	0.08	6.30
		No	0.59	0.97	9.08	0.09	7.67
	c(5, 50):0.005	Yes	0.60	0.98	1.14	0.08	1.55
		No	0.42	0.99	3.03	0.10	2.40
0.95:100:1	c(10, 30):0.00025	Yes	0.68	0.85	4.11	0.09	7.93
		No	0.66	0.80	8.17	0.10	8.66
	c(10, 30):0.005	Yes	0.67	0.86	1.63	0.09	2.96
		No	0.64	0.83	2.99	0.10	3.03
	c(5, 50):0.00025	Yes	0.60	0.95	3.25	0.09	6.15
		No	0.56	0.94	7.34	0.10	7.73
	c(5, 50):0.005	Yes	0.52	0.97	1.02	0.09	1.89
		No	0.39	0.97	2.30	0.11	2.24
0.95:100:3	c(10, 30):0.00025	Yes	0.95	0.93	3.04	0.04	5.69
		No	0.94	0.93	5.43	0.04	6.19
	c(10, 30):0.005	Yes	0.92	0.96	1.13	0.04	2.17
		No	0.90	0.96	1.66	0.04	1.87
	c(5, 50):0.00025	Yes	0.93	0.97	2.30	0.04	4.39
		No	0.91	0.98	4.11	0.04	4.40
	c(5, 50):0.005	Yes	0.78	1.00	0.75	0.04	1.45
		No	0.71	1.00	1.35	0.05	1.39
0.95:50:1	c(10, 30):0.00025	Yes	0.38	0.83	4.05	0.14	5.42
		No	0.35	0.83	8.34	0.15	6.94
	c(10, 30):0.005	Yes	0.37	0.84	1.67	0.14	2.24
		No	0.32	0.86	3.86	0.15	3.13
	c(5, 50):0.00025	Yes	0.18	0.96	4.63	0.16	6.23
		No	0.14	0.97	8.57	0.16	6.82
	c(5, 50):0.005	Yes	0.24	0.94	1.22	0.14	1.65
		No	0.10	0.98	2.43	0.16	1.94
0.95:50:3	c(10, 30):0.00025	Yes	0.80	0.87	4.44	0.07	5.90
		No	0.79	0.85	8.75	0.08	7.77
	c(10, 30):0.005	Yes	0.79	0.90	1.91	0.07	2.55
		No	0.75	0.88	3.26	0.08	2.67
	c(5, 50):0.00025	Yes	0.69	0.95	5.03	0.08	7.50
		No	0.59	0.96	9.93	0.09	8.57
	c(5, 50):0.005	Yes	0.62	0.98	1.25	0.07	1.72
		No	0.43	0.99	3.10	0.10	2.52

Table 1: Means of sensitivity (ratio of included coefficients $\geq .5$), specificity (ratio of excluded coefficients $= 0$), integrated autocorrelation times, root mean square error for estimated coefficients and estimated times per independent sample (in milliseconds, on an AMD Opteron 270)

with an incidence matrix $\boldsymbol{Z}$ for a grouping factor and

$$x_i \overset{\text{i.i.d.}}{\sim} U\left(0, \sqrt{12}\right),\ i = 1, \ldots, n \text{ so that } \operatorname{Var}(\boldsymbol{x}) = 1$$
$$\tilde{b}_g \overset{\text{i.i.d.}}{\sim} t_\nu,\ g = 1, \ldots, \text{no. of groups};$$
$$\boldsymbol{b} = \sigma \frac{\tilde{\boldsymbol{b}} - \operatorname{mean}(\tilde{\boldsymbol{b}})}{\operatorname{sd}(\tilde{\boldsymbol{b}})}$$

with all combinations of the following settings:

- 10 or 100 groups/subjects (i.e $\boldsymbol{b} \in \mathbb{R}^{10}$ or $\mathbb{R}^{100}$)
- with (on average) 5 or 20 observations for each group/subject
- with degrees of freedom $\nu = 1$ or 20 (i.e. Cauchy or approximately Gaussian random effects)

We use scaled and centered random effects $\boldsymbol{b}$ so that the contribution of the random effects to the variability of the linear predictor is constant across replications for the same value of σ and for different values of ν. We compare misclassification rates and estimation error $\|\hat{\boldsymbol{b}} - \boldsymbol{b}\|_2$ between various prior settings for our approach and mixed models fitted with lme4 [Bates and Maechler, 2009] and tested with (restricted) likelihood ratio test.

Linear mixed model

For the linear mixed model, we use

- signal-to-noise-ratio SNR = 1, 5
- random effects scale factor $\sigma = 0$, 0.0625, 0.125, 0.25, 0.5, 0.75, 1

and balanced data, in addition to the settings described above. We generate 100 data sets for each combination of settings.

Inclusion or exclusion of the random intercept term in the LMM is based on the p-value of an exact restricted likelihood ratio test (RLRT) for $H_0 : \sigma^2 = 0$ with significance level $\alpha = .05$ as implemented in RLRsim [Scheipl, 2010a, Scheipl et al., 2008]. We consider the random intercept included in the spikeSlabGAM-models if the Rao-Blackwellized estimate of the posterior mean of $P(\gamma_b = 1)$ is greater than .5. Figure 13 shows error rates (top left: false positive or type I error for $\sigma = 0$, top right: false negative or type II error) and root mean square estimation errors for the random intercept model for Gaussian responses. Type I and type II error rates for the hyperparameter configurations considered here are fairly close to those of the RLRT with significance level $\alpha = .05$ (black lines). As in the other simulations, a smaller v_0 (red, blue symbols) yields less conservative models, because the threshold an effect has to cross before the associated hypervariance is sampled from the "slab" and not from the "spike" decreases. Type II error rates are insensitive towards the different prior combinations for smaller sample sizes and low SNR.

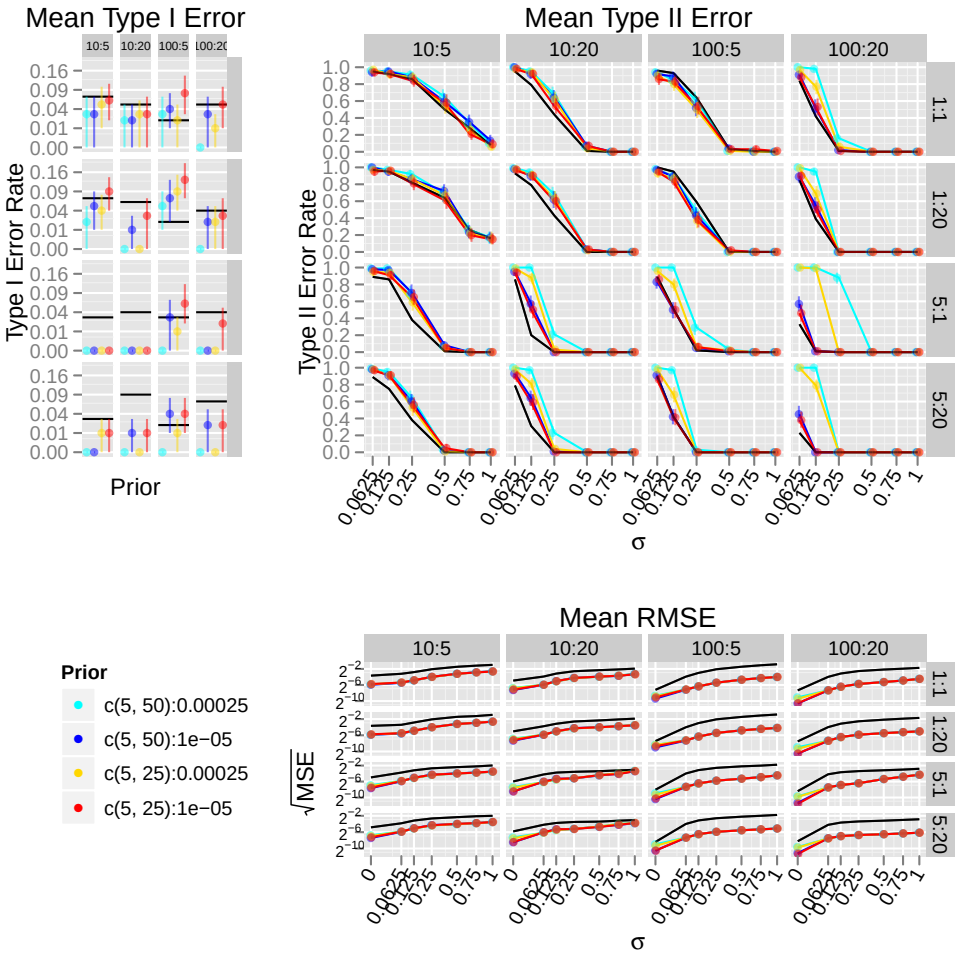

Figure 13: Mean type I / type II err rates and $\sqrt{\text{MSE}}$ for linear mixed models with a random intercept.
Rows correspond to the different combinations of SNR and degrees of freedom ν, top two rows are for SNR $= 1$. Columns correspond to the different combinations of number of groups/subjects and observations per group/subject, two rightmost columns are for 10 groups/subjects. Left graph gives type I error for $\sigma = 0$, right graph gives type II error rates for $\sigma > 0$. Graph on the lower right gives mean estimation error $\sqrt{\text{MSE}} = \sqrt{\|\hat{\boldsymbol{b}} - \boldsymbol{b}\|^2}$. Solid black lines line give error rates and RMSE for the LMM (based on the p-value of a restricted LR-test with $\alpha = .05$). Vertical axis for type I error is on $\sqrt{}$-scale, vertical axis for RMSE is on $\log_2$-scale. Error bars show 95% CIs for mean error rates.

Across all settings, estimation error for the LMM is markedly larger than for peNMIG and fairly stable across the different priors. Estimation error for $\sigma = 0$, however, is much lower for $v_0 = 0.00001$ since it imposes stronger shrinkage than $v_0 = 0.00025$.

Mixed model with binary response

For the generalized linear mixed model, binary responses $\boldsymbol{y}$ are generated from

$$y_i \sim B\left(n = 1, p = \left(1 + \exp\left(-\eta_i\right)\right)^{-1}\right)$$

with

- random effects scale factor $\sigma = 0$, $0.125, 0.25$, 0.5, 0.75, 1, 1.5
- balanced groups or unbalanced groups with relative group sizes drawn from a Dirichlet distribution with concentration parameter $\boldsymbol{\alpha} = (5, \ldots, 5)'$

and the other settings as described at the beginning of this Section. Inclusion or exclusion of the smooth term in the GLMM is based on the p-value of a likelihood ratio test for $H_0 : \sigma^2 = 0$ with significance level $\alpha = 0.15$. The reference distribution for this test was determined by a parametric bootstrap for each dataset. We generate 50 data sets for each combination of settings. Figure 14 shows error rates (top left: false positive or type I error for $\sigma = 0$, top right: false negative or type II error) and root mean square estimation errors for the random intercept model for Binomial responses. Type I error rates for the peNMIG model are larger for small group size (first and third column) and this difference is more pronounced in the balanced settings (second and fourth row) than in the unbalanced settings. For those settings where peNMIG and the bootstrap LRT have similar type I error rates, their type II error rates are similar as well, and for all settings the slope of type II error rates for peNMIG is similar to that of the bootstrap LRT. Surprisingly, a smaller v_0 (red, blue symbols) does not yield less conservative models for many of the settings. Across all settings, estimation error for the GLMM is markedly larger than for peNMIG and practically indistinguishable for the different priors.

The simulation results for LMM and GLMM suggest that peNMIG's model selection behavior for random effects is similar to that of the (restricted) likelihood ratio test for a broad variety of settings, but peNMIG's estimation of the random effects is much better than that produced by the conventional ridge-type shrinkage of the frequentist mixed model with Gaussian random effects.

5.4 Univariate Smoothing for Gaussian response

We investigate the properties of the peNMIG prior in terms of function selection for both randomly generated and fixed functions.

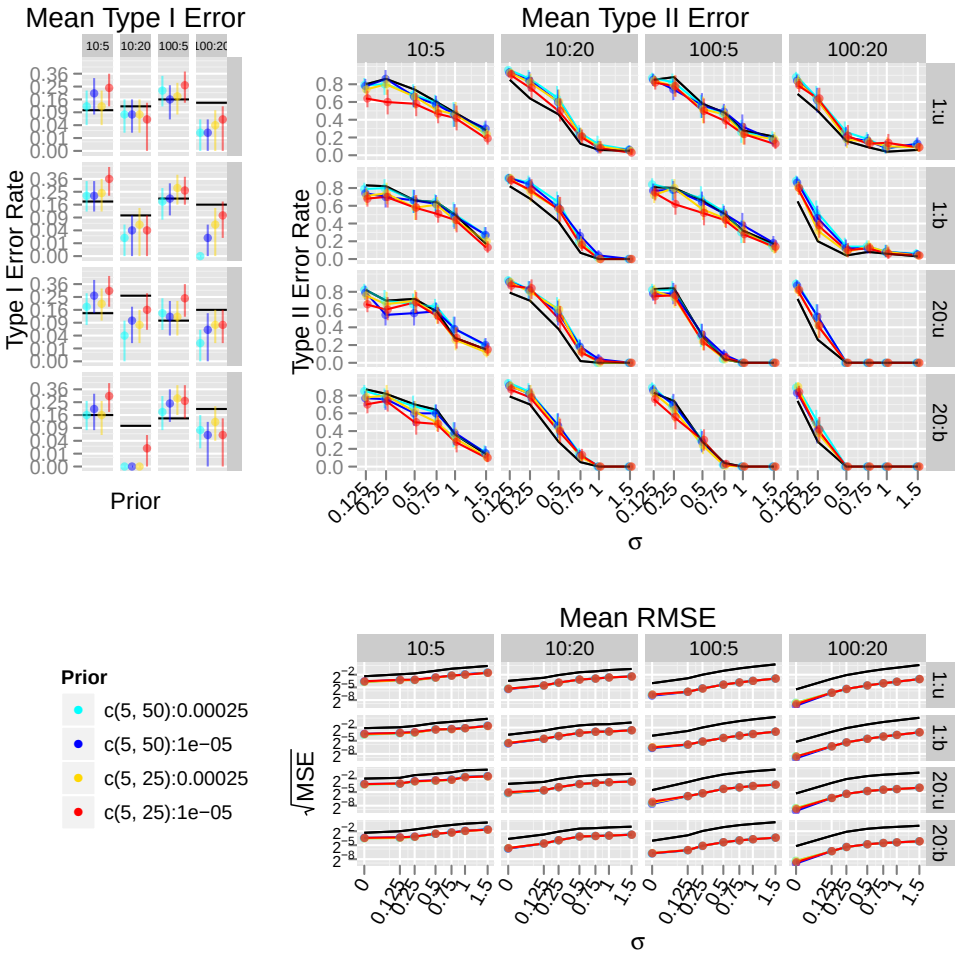

Figure 14: Mean type I / type II error rates and $\sqrt{\text{MSE}}$ for mixed models with a random intercept and binary response.
Rows correspond to the different combinations of balance ("u" is unbalanced, "b" is balanced) and degrees of freedom ν, top two rows are for $\nu = 1$. Columns correspond to the different combinations of number of groups/subjects and observations per group/subject, two rightmost columns are for 10 groups/subjects. Left graph gives type I error for $\sigma = 0$, right graph gives type II error rates for $\sigma > 0$. Graph on the lower right gives mean estimation error $\sqrt{\text{MSE}} = \sqrt{\|\hat{\boldsymbol{b}} - \boldsymbol{b}\|^2}$. Solid black lines line give error rates and RMSE for the GLMM (based on the p-value of a LR test with $\alpha = .15$). Vertical axis for type I error is on $\sqrt{}$-scale, vertical axis for RMSE is on $\log_2$-scale. Error bars show 95% CIs for mean error rates.

We compare inclusion probabilities and misclassification rates for peNMIG with various hyperparameter configurations to boosting with separate base learners for the linear and smooth parts of the function with mboost and to additive models (AM) in mixed model representation fitted with amer [Scheipl, 2010b]. Inclusion or exclusion of a smooth term in the AM is based on the p-value of an RLRT for $H_0 : \sigma^2 = 0$ with $\alpha = .05$ as implemented in RLRsim [Scheipl, 2010a, Scheipl et al., 2008]. Ten-fold cross validation on the training data is employed to determine the optimal stopping iteration for mboost and a baselearner is included in the model if it is selected in at least half of the cross-validation runs up to the stopping iteration. Smooth terms are included in the spikeSlabGAM-models if the Rao-Blackwellized posterior mean of $P(\gamma = 1)$ is greater than .5.

Randomly generated functions

We investigate the properties of our approach first on data from a very basic data-generating process for a simple spline model:

- $\boldsymbol{\eta} = \boldsymbol{x} + \boldsymbol{Z}(\boldsymbol{x})\boldsymbol{b}$; $\boldsymbol{Z}(\boldsymbol{x})$ is the penalized part of a B-spline basis for covariate $\boldsymbol{x}$ with a difference penalty of order 2.
- $\boldsymbol{b} \sim \sigma N(\boldsymbol{\mu}, \boldsymbol{I}_d)$, $\boldsymbol{\mu}$ is drawn from $\{-1, 1\}^d$.

We use the following settings for the simulation:

- number of observations: $n = 50, 500$
- signal-to-noise-ratio SNR $= 0.5, 5$
- dimension of spline basis: $d_s = 5, 20$
- degree of nonlinearity: $\sigma^2 = 0, 0.125, 0.25, 0.375, 0.5$
- 50 replications

For $\sigma^2 = 0$, the function to be estimated is linear, so the correct model is one without a smooth term. Results for this data generating process are shown in Figure 16. Figure 15 shows 10 realizations of simulated functions $\boldsymbol{x} + \boldsymbol{Z}(\boldsymbol{x})\boldsymbol{b}$ for the various settings.

Fixed functions

We also investigate the properties of our approach with a data-generating process (DGP) based on nonrandom nonlinear functions:

- $\eta = \boldsymbol{x} + \sigma f(x)$
- $f(x) = \begin{cases} (2x - 1.5)^2/3 & \text{(quadratic)} \\ (\pi \sin(2\pi x))/11 & \text{(sinus)} \\ (\phi((x - 0.2)/0.12) - \phi((x - 0.7)/0.055)) & \text{(bumpy)} \end{cases}$

  ($\phi(\cdot)$ is the standard normal density.)

We use the following settings for the simulation:

- number of observations: $n = 50, 500$
- signal-to-noise-ratio SNR $= 0.5, 5$
- degree of nonlinearity: $s = 0, 0.25, 0.5, 0.75, 1$
- 50 replications

For $\sigma = 0$, the function to be estimated is a simple line, so the correct model is one without a smooth term. Figure 15 shows the shape of the 3 functions for varying d. We use 10 basis functions to estimate the functions.

Figures 16 and 17 show type I and type II error rates along with square root of the mean square error $\|\boldsymbol{\eta} - \hat{\boldsymbol{\eta}}\|^2$ for the various priors, additive models fit with amer and tested with exactRLRT (solid lines) and component-wise boosting fit with mboost. Selection via component-wise boosting is extremely anti-conservative, with type I error rate between 60% and 95% and type II error rates close to 0 across all settings, and comparatively large prediction error especially for strong nonlinearity.

Inclusion probabilities for spikeSlabGAM are heavily influenced by the prior settings: Note that $(a_\tau, b_\tau) = (10, 30)$ implies smaller hypervariances than $(a_\tau, b_\tau) = (5, 50)$ and thus less regularization of the function estimates, so the higher inclusion rates for $(a_\tau, b_\tau) = (10, 30)$ are expected. Since smaller values of v_0 imply stronger regularization if the hypervariance is sampled from the "spike", the odds of sampling from the "spike" are smaller and thus the smaller values of v_0 (i.e. lighter shades in figs. 16, 17) are less conservative and quicker to include smooth terms in the model (i.e. sample from the "slab") – the smaller v_0, the smaller is the threshold an effect has to cross in order to be included in the model.

Compared to function selection based on the RLRT with nominal $\alpha = .05$ – note that model selection via AIC corresponds to an RLRT with $\alpha = .05$ in this context [Greven, 2007, p. 104] – our approach is more conservative for almost all of the considered settings and priors. Exceptions occur for priors with small v_0 in settings with low-dimensional basis, small n and/or low signal-to-noise ratios. In those settings, the prior influence is much stronger and there is not enough information in the data to move $P(\gamma = 1)$ far away from its prior mean of .5 in many cases.

Correspondingly, type II error rates are mostly higher than those for the RLRT, with exceptions for those settings and priors that are less conservative. There is only one combination of prior and setting in which our approach dominates the RLRT in terms of misclassification: For randomly generated functions with $n = 500, \text{SNR} = .5, d = 5$ and prior $(10, 30) : .005$, both type I and type II error rates are lower than those of the RLRT. For most settings and priors, type II error rates decrease about as fast as those of the RLRT, but on a higher absolute level. This reflects the fact that the model selection implemented in spikeSlabGAM selects "relevant" terms and not "significant" terms. The threshold of relevance depends on (a_τ, b_τ) and v_0.

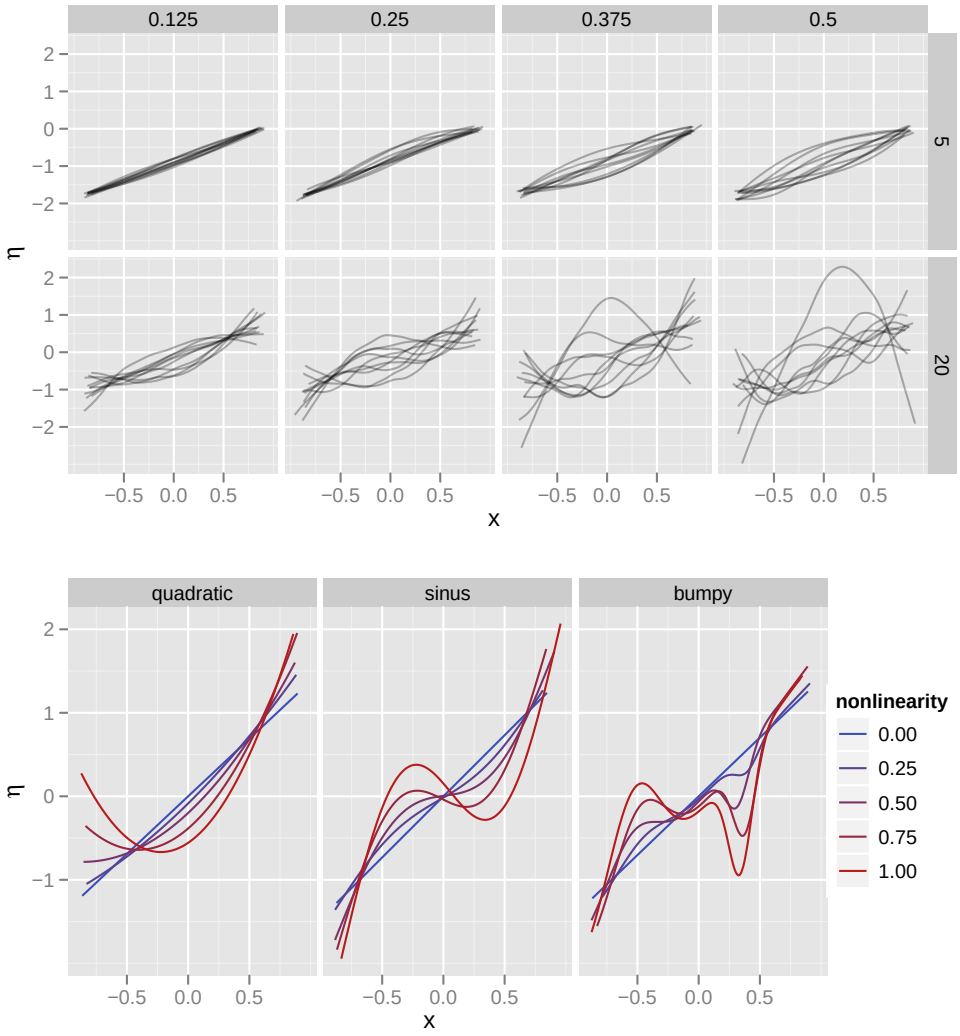

Figure 15: True linear predictor for univariate smoothing simulations.
Upper graph displays randomly generated functions: Upper row for 5 basis functions, lower row for 20 basis functions. Columns correspond to the different settings of $\sigma^2 > 0$.
Bottom graph displays true linear predictors for the fixed functions: Columns correspond to the 3 different functions, line color indicates the value of the nonlinearity parameter s.

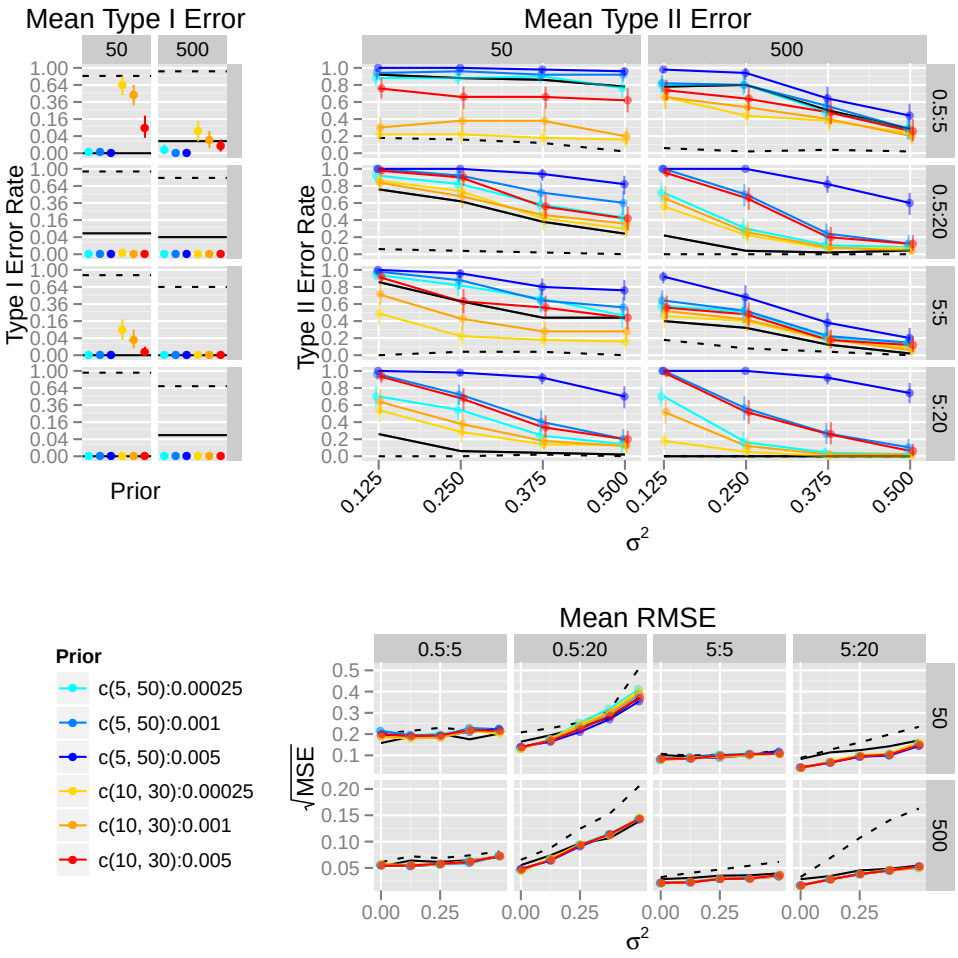

Figure 16: Mean type I / type II error rates and $\sqrt{\text{MSE}}$ for randomly generated functions.
Left graph gives type I error for $\sigma^2 = 0$, right graph gives type II error rates for $\sigma^2 > 0$. Left column in each graph for $n = 50$, right column for $n = 500$. Upper two rows for SNR $= .5$ with $d_s = 5, 20$, lower two for SNR $= 5$. Graph on the lower right gives mean prediction $\sqrt{\text{MSE}}$. Solid black lines line gives error rates for the GAM (based on the p-value of a restricted LR-test with $\alpha = .05$), dashed black line for mboost. Vertical axis for type I error is on $\sqrt{}$-scale. Error bars show 95% CIs for mean error rates.

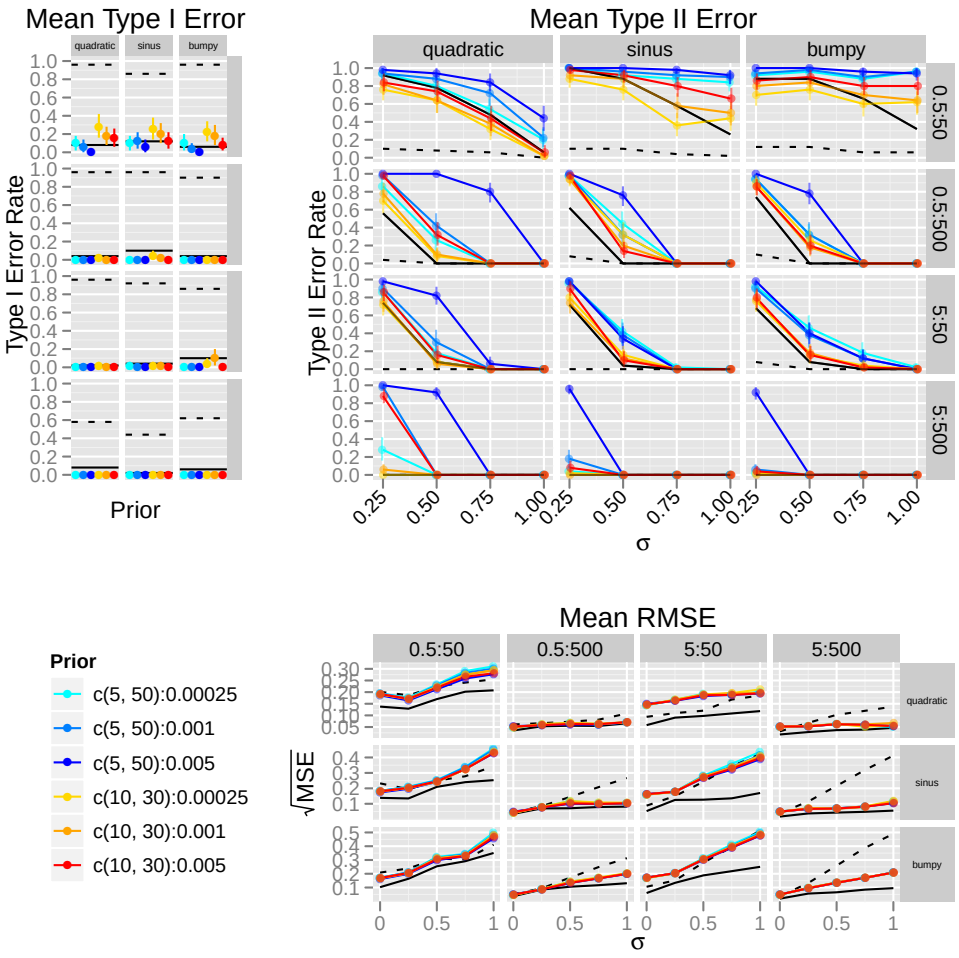

Figure 17: Mean type I / type II error rates and $\sqrt{\text{MSE}}$ for fixed functions. Rows correspond to the three different functions. Left graph gives type I error for $\sigma = 0$, right graph gives type II error rates for $\sigma > 0$. Columns show results for the different functions. Top two rows for SNR $= .5$ with $n = 5,500$, bottom rows for SNR $= 5$. Graph on the lower right gives mean prediction $\sqrt{\text{MSE}}$. Solid black lines line gives error rates for the GAM (based on the p-value of a restricted LR-test with $\alpha = .05$), dashed black line for mboost. Vertical axis for type I error is on $\sqrt{}$-scale. Error bars show 95% CIs for mean error rates.

The graphs on the bottom right of figs. 16 and 17 shows that the larger type II error rates do not necessarily mean higher estimation errors. For randomly generated functions, the model averaging implicit in our procedure fits the data as least as good as the frequentist AM in this context, and seems to perform much better than component-wise boosting. Average estimation errors for fixed functions are mostly smaller than those of component-wise boosting, but larger than those for the additive model. Across all settings, estimation errors are much more robust against the different prior settings than model selection.

We discussed the shape of the multivariate prior for grouped coefficients in Section 3.4 and noted that it places more mass on coefficient vectors with many entries of a similar size. The specific fixed functions we used were chosen since the *true* coefficient vector for the penalized basis functions for the quadratic function lies on the bisecting angle, i.e. all entries have the same value, while the entries of the *true* coefficient vectors for both the sinus and the bumpy functions have strongly varying magnitudes. Consequently, we expected performance for the quadratic function to be much better than the performance for the other two fixed functions. It is reassuring to see that the relative performance for the quadratic function is very similar to that for the sinus and bumpy functions, even for settings where the information content in the likelihood is fairly weak (low SNR, low n) and the potential for prior-data conflict to distort the fit is correspondingly large.

5.5 Generalized Additive Models

In the following Sections 5.5.1 and 5.5.2, we compare the performance of peNMIG in sparse (generalized) additive models to that of component-wise boosting [Hothorn et al., 2010] in terms of predictive MSE and complexity recovery. As a reference, we also fit a conventional GAM (as implemented in mgcv [Wood, 2008]) based on the "true" formula (i.e. a model without any of the "noise" terms), which we subsequently call the "oracle"-model. For Gaussian responses only, we also compare our results to those from ACOSSO [Storlie et al., 2009]. ACOSSO is not able to fit non-Gaussian responses.

We supply separate base learners for the linear and smooth parts of covariate influence for the component-wise boosting in order to compare complexity recovery between boosting and our approach. We use 10-fold cross validation on the training data to determine the optimal stopping iteration for mboost and count a baselearner as included in the model if it is selected in at least half of the cross-validation runs up to the stopping iteration. BIC is used to determine the tuning parameter for ACOSSO. We were unable to compare our approach to the closely related one described in [Reich et al., 2009], which is implemented for Gaussian responses, since the available R implementation is impractically slow.

For both Gaussian responses (Section 5.5.1) and Poisson responses (Section 5.5.2), the data generating process has the following structure:

- We define 4 functions

  - $f_1(x) = x$,
  - $f_2(x) = x + \frac{(2x-2)^2}{5.5}$,
  - $f_3(x) = -x + \pi \sin(\pi x)$,
  - $f_4(x) = 0.5x + 15\phi(2(x - .2)) - \phi(x + 0.4)$, where $\phi()$ is the standard normal density function,

  which enter into the linear predictor. Note that all of them have (at least) a linear component.

- We define 2 scenarios:
  - a "low sparsity" scenario: Generate 16 covariates, 12 of which have non-zero influence: the true linear predictor is $\eta = f_1(x_1) + f_2(x_2) + f_3(x_3) + f_4(x_4) + 1.5(f_1(x_5) + f_2(x_6) + f_3(x_7) + f_4(x_8)) + 2(f_1(x_9) + f_2(x_{10}) + f_3(x_{11}) + f_4(x_{12}))$
  - "hi sparsity" scenario: Generate 20 covariates, 4 of which have non-zero influence: $\eta = f_1(x_1) + f_2(x_2) + f_3(x_3) + f_4(x_4)$
- The covariates are either
  - $\overset{\text{i.i.d.}}{\sim} U[-2, 2]$ or
  - from an AR(1) process with correlation $\rho = 0.7$.
- We simulate 50 replications for each combination of the various settings.

We compare 9 different prior specifications:

- $(a_\tau, b_\tau) = (5, 25)$, $(10, 30)$, $(5, 50)$
- $v_0 = 0.00025, 0.005, 0.01$

Predictive MSE is evaluated on test data sets with 5000 observations. Complexity recovery, i.e. how well the different approaches select covariates with true influence on the response and remove covariates without true influence on the response is measured in terms of accuracy, defined as the number of correctly classified model terms (true positives and true negatives) divided by the total number of terms in the model. I.e. for the "low sparsity" scenario, the full model potentially has 32 terms (linear terms and basis expansions/smooth terms for each of the 16 covariates), only 21 of which are truly non-zero (the linear terms for the first 12 covariates plus the 9 basis expansions of the covariates not associated with the linear function $f_1()$). Accuracy in this scenario would then be determined as the sum of the correctly included model terms plus the correctly excluded model terms, divided by 32.

5.5.1 Gaussian response

In addition to the basic structure of the data generating process described in the previous Section, the data generating process for the Gaussian responses has the following properties:

- signal-to-noise-ratio SNR $= 5, 20$
- number of observations: $n = 200, 1000$

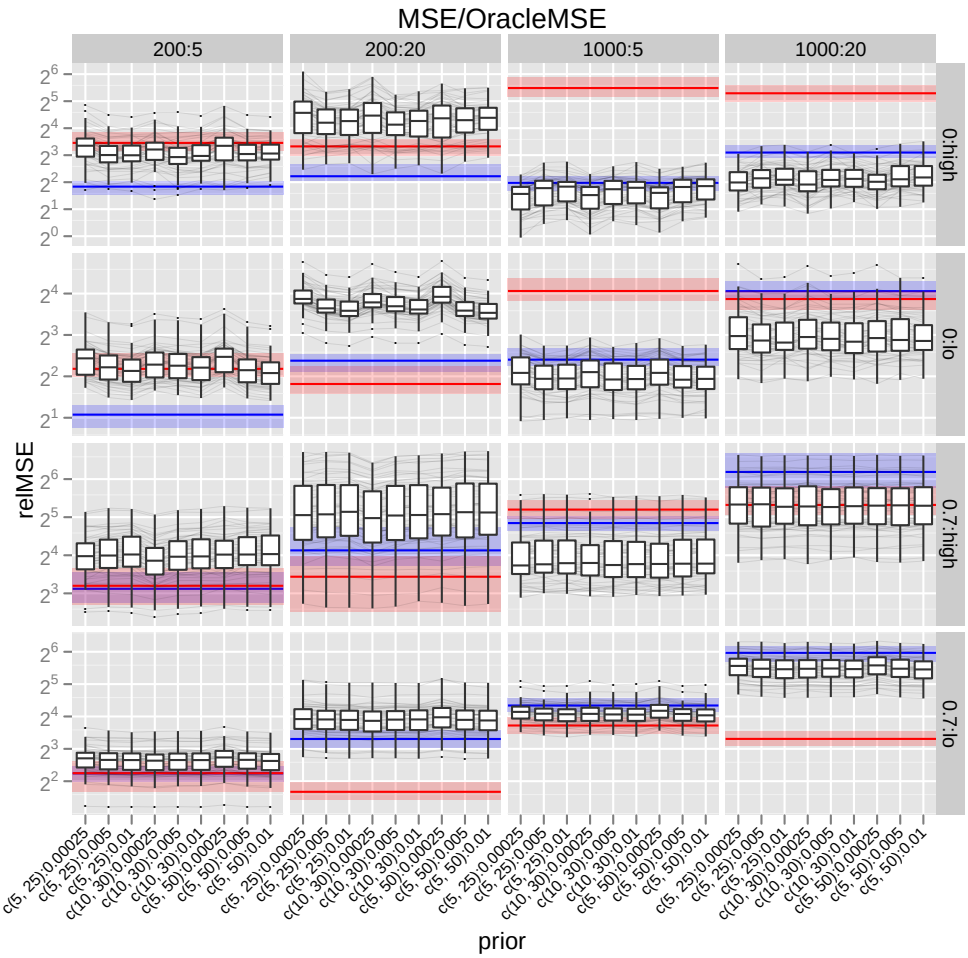

Figure 18: Prediction MSE divided by oracle MSE for Gaussian response: White boxplots show results for the different prior settings, blue and red symbols show results for mboost and ACOSSO, respectively: Shaded region gives IQR, line represents median. Dark grey lines connect results for the same replication.
Columns from left to right: 200 obs. with SNR=5, 20; 1000 obs. with SNR=5, 20. Rows from top to bottom: uncorrelated obs. with sparse and unsparse predictor, correlated obs. with sparse and unsparse predictor. Vertical axis is on binary log scale.

Figure 18 shows the mean squared prediction error divided by the one achieved by the "oracle"-model. Predictive performance is very robust against the different prior settings. Different prior settings also behave similarly within replications, as shown by the mostly parallel grey lines. Predictions for $N = 1000$ (two rightmost columns) are mostly better than and at least equal to

the results for both boosting and ACOSSO, with the exception of the setting with unsparse predictor and correlated observations (lowest right panels), where ACOSSO outperforms our approach. Predictions for the small sample case ($N = 200$, two leftmost columns) are similar but generally less precise than either boosting or ACOSSO, especially for high SNR and uncorrelated observations (top half of second column). Figure 19 shows the proportion of

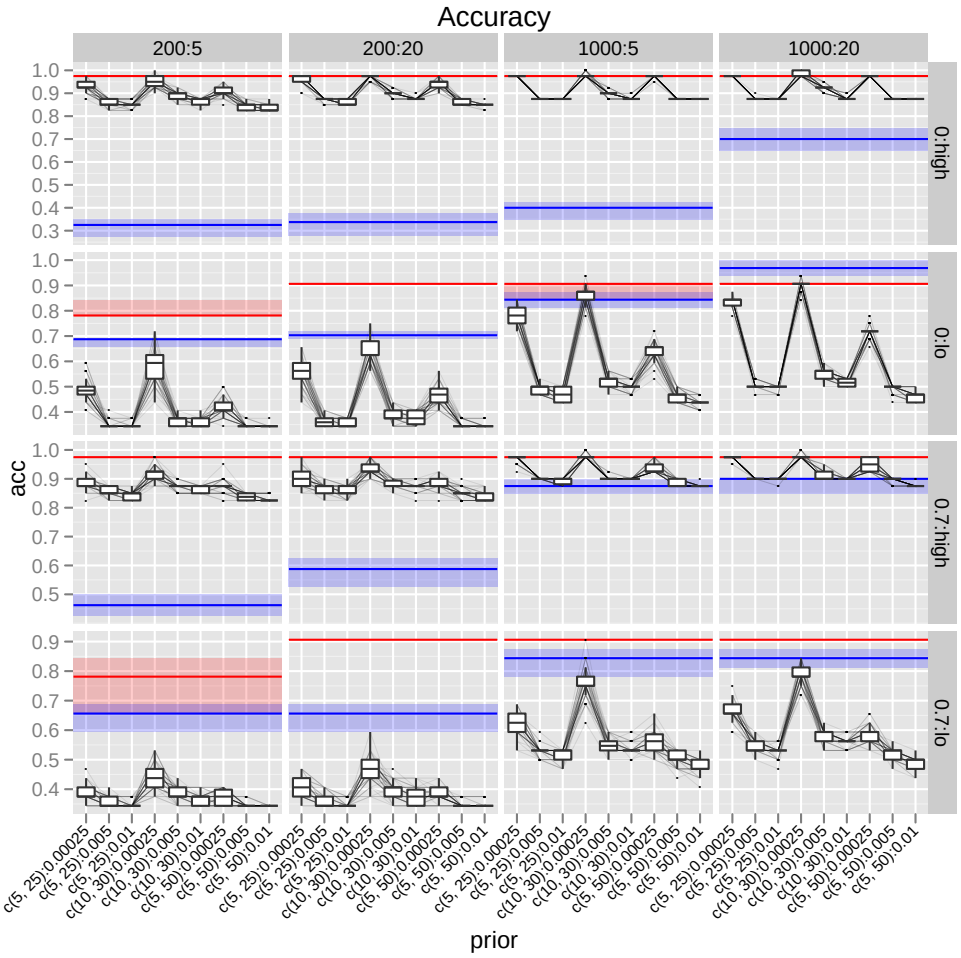

Figure 19: Complexity recovery for Gaussian response: proportion of correctly included and excluded model terms.
White boxplots show results for the different prior settings, blue and red symbols show results for mboost and ACOSSO, respectively: Shaded region gives IQR, line represents median. Dark grey lines connect results for the same replication.
Columns from left to right: 200 obs. with SNR=5, 20; 1000 obs. with SNR=5, 20. Rows from top to bottom: uncorrelated obs. with sparse and unsparse predictor, correlated obs. with sparse and unsparse predictor.

correctly included and excluded terms (linear terms and basis expansions) in the estimated model. Estimated inclusion probabilities are very sensitive to

v_0 and comparatively robust against (a_τ, b_τ). Accuracy is consistently much lower than for ACOSSO. However, a direct comparison with ACOSSO is not entirely appropriate because ACOSSO does not differentiate between smooth and linear terms, while mboost and our approach do. Therefore ACOSSO solves a less difficult problem. The accuracy of peNMIG is always better than mboost for the sparse settings (1st and 3rd rows) because the specificity of our approach is 1 across settings, regardless of the prior (!), while mboost mostly achieves only very low specificity, but very high sensitivity.

5.5.2 Poisson response

In addition to the basic structure of the data generating process described in the previous section, the data generating process for the Poisson responses has the following properties:

- number of observations: $n = 500, 2000$
- responses are generated with overdispersion:
  $y_i \sim Po\left(s_i \exp(\eta_i)\right)$; $s_i \sim U[0.66, 1.5]$

Figure 20 shows the mean squared prediction error (on the scale of the linear predictor) divided by the one achieved by the "oracle"-GAM. Predictive performance is very robust against the different prior settings. Different prior settings also behave similarly within replications, as shown by the mostly parallel grey lines. Predictions for uncorrelated responses (top 2 rows) are mostly more precise than mboost, especially so for the sparse setting with uncorrelated responses (top row). Note that our approach even seems to improve on the oracle method for about half of the replications in the uncorrelated, unsparse setting with $n = 500$ (second row, first column) with a relative prediction MSEs below 1. Predictions for correlated responses (bottom 2 rows) are generally less precise than mboost, especially for the sparse setting (third row). Figure 21 shows the proportion of correctly included and excluded terms (linear terms and basis expansions) in the estimated models. Estimated inclusion probabilities are sensitive to v_0 and comparatively robust against (a_τ, b_τ). The smaller value for v_0 tends to perform better in the unsparse settings (rows 2 and 4) since it forces more terms into the model (higher sensitivity, lower specificity) and vice versa for the sparse setting and the larger v_0. Complexity recovery is usually more stable across the different settings and priors for our approach than for boosting, especially for uncorrelated covariates and a sparse predictor (top row). The constant accuracy for mboost in the low sparsity scenario with uncorrelated responses (second row) is due to its very low specificity: It includes practically all model terms all the time.

The simulations for generalized additive models show that the proposed peNMIG-Model is competitive in terms of estimation accuracy and confirms that estimation results are robust against different hyperparameter configurations even in fairly complex models. Model selection is more sensitive towards hyperparameter configurations, especially v_0. The discovery rate of

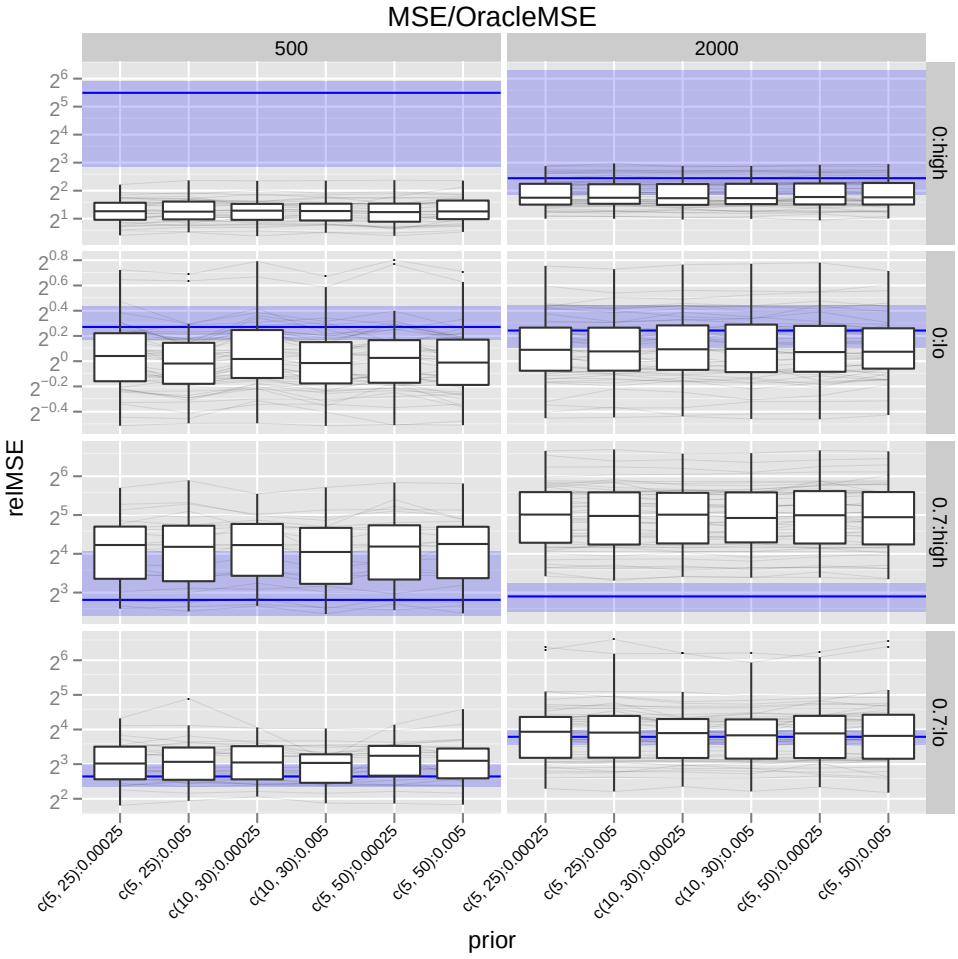

Figure 20: Prediction MSE divided by oracle MSE (on the scale of the linear predictor):
White boxplots show results for the different prior settings, blue symbols show results for mboost. Shaded region gives IQR, line represents median. Dark grey lines connect results for the same replication.
Columns from left to right: 500 obs., 2000 obs. Rows from top to bottom: uncorrelated obs. with sparse and unsparse predictor, correlated obs. with sparse and unsparse predictor. Vertical axis is on binary log scale.

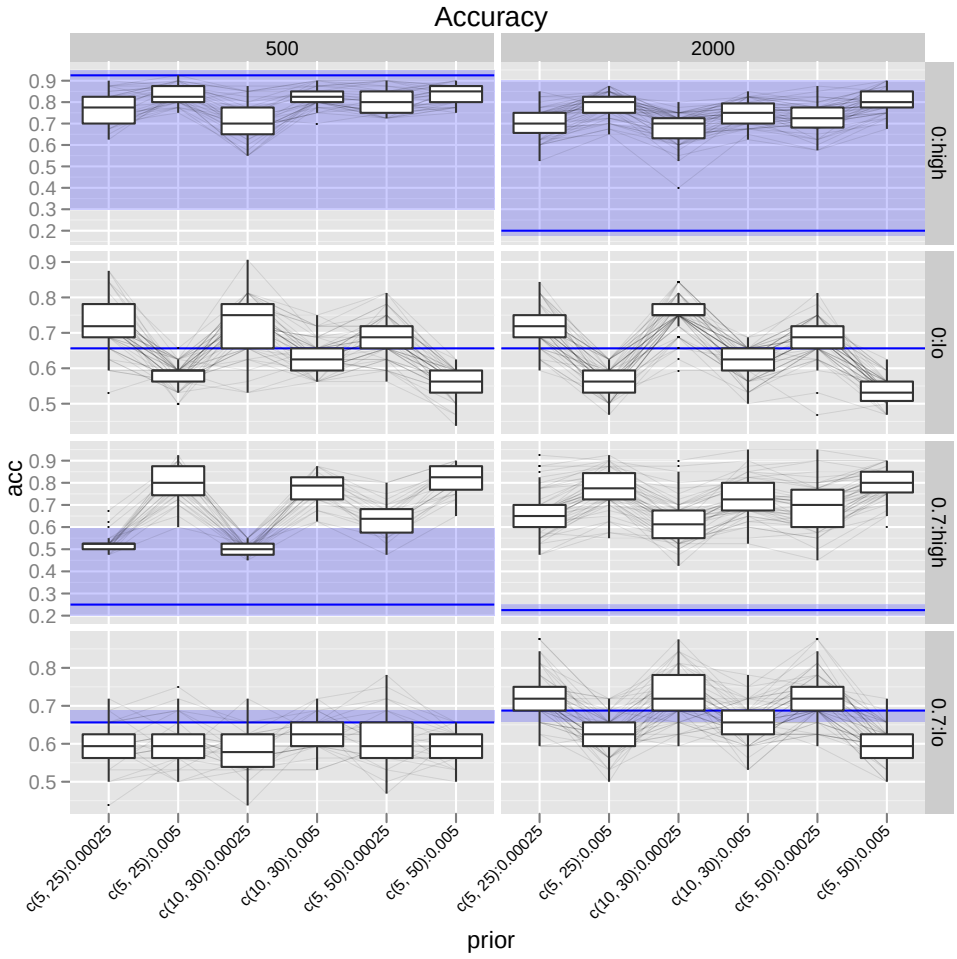

Figure 21: Complexity recovery for poisson response: proportion of correctly included and excluded model terms.
White boxplots show results for the different prior settings, blue symbols show results for mboost: shaded region gives IQR, line represents median. Dark grey lines connect results for the same replication.
Columns from left to right: 500 obs., 2000 obs. Rows from top to bottom: uncorrelated obs. with sparse and unsparse predictor, correlated obs. with sparse and unsparse predictor.

true non-zero model terms was low for Gaussian responses, and high for Poisson responses.

We are not aware of any other SSVS implementations for variable selection in additive models with non-Gaussian responses that were available for benchmarking, but the performance of peNMIG seems to be very competitive to that of component-wise boosting.

6 Applications

6.1 UCI Binary Classification Data

We use a collection of 21 data sets for binary classification from the UCI Machine Learning Repository [Asuncion and Newman, 2007]. Figure 22 gives

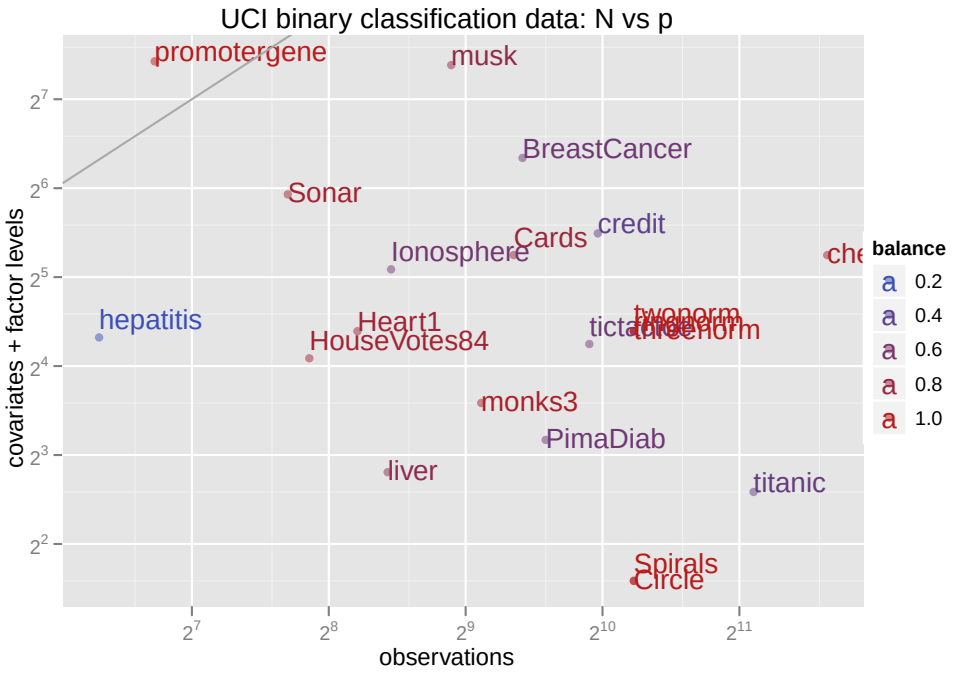

Figure 22: Characteristics of UCI data sets: number of observations versus number of features.
"Balance" is the ratio between the number of observations in the larger class and the number of observations in the smaller class, i.e. it is 1 if the data set is balanced. `promotergene` is the only dataset we consider that has more parameters than observations before accounting for spline basis expansions.

an overview of the datasets we use and their characteristics. The vertical axis gives the number of covariates and different factor levels, the horizontal axis gives the number of (complete) observations. Most of the datasets contain a mixture of continuous and factor variables. We do not consider any interactions, only linear and smooth main effects. We evaluate prediction

performance based on the deviance values for a 20-fold cross validation on each dataset. Predictive deviance $\bar{D}$ is defined as twice the average negative log likelihood $\bar{D} = -2/n_P \sum_{i=1}^{n_P} L(y_{P,i}, \hat{\eta}_{P,i})$ in the test sample where $\boldsymbol{y}_P$ and $\hat{\boldsymbol{\eta}}_P$ are the out-of-sample responses and estimated linear predictors for the test sample. The size of the test sample is n_P. As for the experiments with simulated data, we use component-wise boosting with separate base learners for the linear and smooth parts of covariate influence and compare prediction performance and complexity of the boosting models to our approach.

We brutally preprocess the data in an automated fashion in order to preempt possible numerical problems: All covariates with less than 6 unique values are coded as factor variables. All numeric covariates are scaled to the unit interval $[0, 1]$ first, followed by taking the logarithm of the covariate values (plus an offset of .1) if skewness is greater than 2 or taking the logarithm of 1.1 minus the covariate value if skewness is below -2. All numeric covariates (transformed or not) are then standardized to have mean 0 and standard deviation 1. All incomplete observations are removed.

We evaluate our approach for two model building scenarios: For the first one, we perform an automated preselection procedure to generate model formulas based on the following heuristic, which roughly follows ideas developed by Harrell [2001]:

1. Determine the "available degrees of freedom" for the smooth terms by dividing the number of observations in the smaller class by 3 and subtracting the sum of the number of levels of all factor variables in the data.

2. - if the available degrees of freedom are larger than 4 times the number of numeric covariates, assign a spline expansion with 10 basis functions to each numerical covariate. You're done.
   - if not go to next step

3. - split all numerical covariates by quartile
   - perform χ^2-tests of association of the resulting 5-level factors with the response
   - sort numerical covariates by decreasing strength of association (as measured by the p-value of the χ^2-test)

4. starting with the covariate with the strongest association, assign spline expansions with 5 basis functions to the numerical covariates and subtract 5 "available degrees of freedom" until no more degrees of freedom are left

5. if any numerical covariates remain after all available degrees of freedom are spent, they enter the model as simple linear terms.

This approach results in model specifications below the maximum complexity for datasets `credit`, `Cards`, `Heart1`, `Ionosphere`, `hepatitis`, `Sonar` and `musk`.

In the second approach, we assign a spline expansion with 5 basis functions to all numerical covariates regardless of the number of predictors and observations, leading to a more difficult estimation and selection problem in data sets with large p and small n.

Models with preselected function terms

We show results for combinations of $(a_\tau, b_\tau) = (10, 50), (5, 25)$ and $v_0 = 0.005, 0.00025$. We use a uniform prior $w \sim \text{Beta}(1, 1)$. MCMC chains are run with a burn-in of 1000 iterations, followed by a sampling phase of 12000 iterations, of which we save every fourth.

Figure 23 shows the achieved predictive performance for the first model building strategy for the 21 datasets. Note that the performance of our approach is more variable than mboost's. Our approach achieves lower median deviances in most datasets. Predictive performance seems to be very robust against different hyperparameter settings, even for large p/N. With the exceptions of `Spirals`, `PimaDiab`, `ringnorm`, `threenorm`, and `twonorm`, our approach yields more accurate predictions in the majority of cross-validation folds.

To investigate the parsimony of the fitted models, i.e. whether equivalent or better prediction can be achieved by simpler models, we plot the differences in predictive deviances versus the difference in the proportion of potential model terms included in the models in Figure 24. Larger values on the vertical axis indicate smaller deviance for our approach, and larger values on the horizontal axis indicate a sparser fit for our approach.

For dataset `threenorm` our approach tends to yield more complex models with larger deviance. For datasets `credit`, `Cards`, `Ionosphere`, `promotergene` and `Sonar`, our approach predicts more accurately than boosting with (much) smaller model complexity. Neither absolute performance nor performance relative to boosting seems to be tied to any of the easily observable characteristics of the data sets (i.e. p, N, p/N, balancedness).

No clear picture emerges for the different priors: As expected, a smaller v_0 (green and blue dots) tends to yield larger models, i.e. datasets `credit`, `Sonar`, and results are more sensitive towards v_0 than towards (a_τ, b_τ). Note that Figure 24 does not include datasets `Spirals`, `tictactoe`, `twonorm` and `titanic` because there were no differences in sparsity. For both `titanic` and `tictactoe`, prediction was much better with our approach, while prediction for `Spirals` and `twonorm` was worse (see fig. 23). Table 2 gives the median deviances and AUCs (area under the ROC-curve) for the different datasets and priors.

Models without preselection

We use the second model-building strategy and repeat the analysis without restricting the number of smooth terms for data sets `credit`, `Cards`, `Heart1`, `Ionosphere`, `hepatitis`, `Sonar` and `musk` (For all other data sets, the model without preselection is the same as the one with preselection.). We

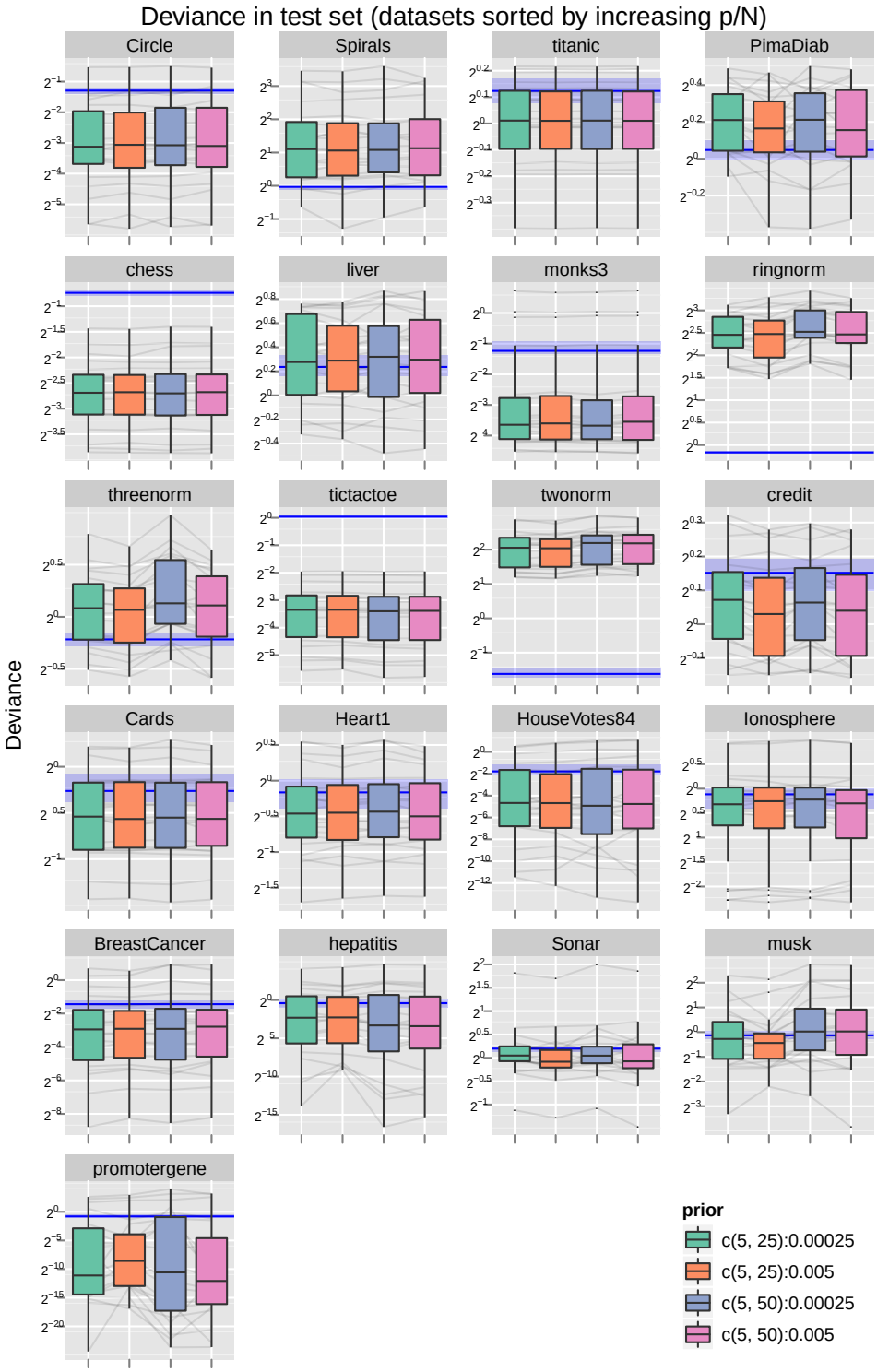

Figure 23: UCI data I: Predictive Deviances for 20-fold crossvalidation. Boxplots show results for the different prior settings, blue symbols show results for mboost: shaded region gives IQR, line represents median. Dark grey lines connect results for the same fold.

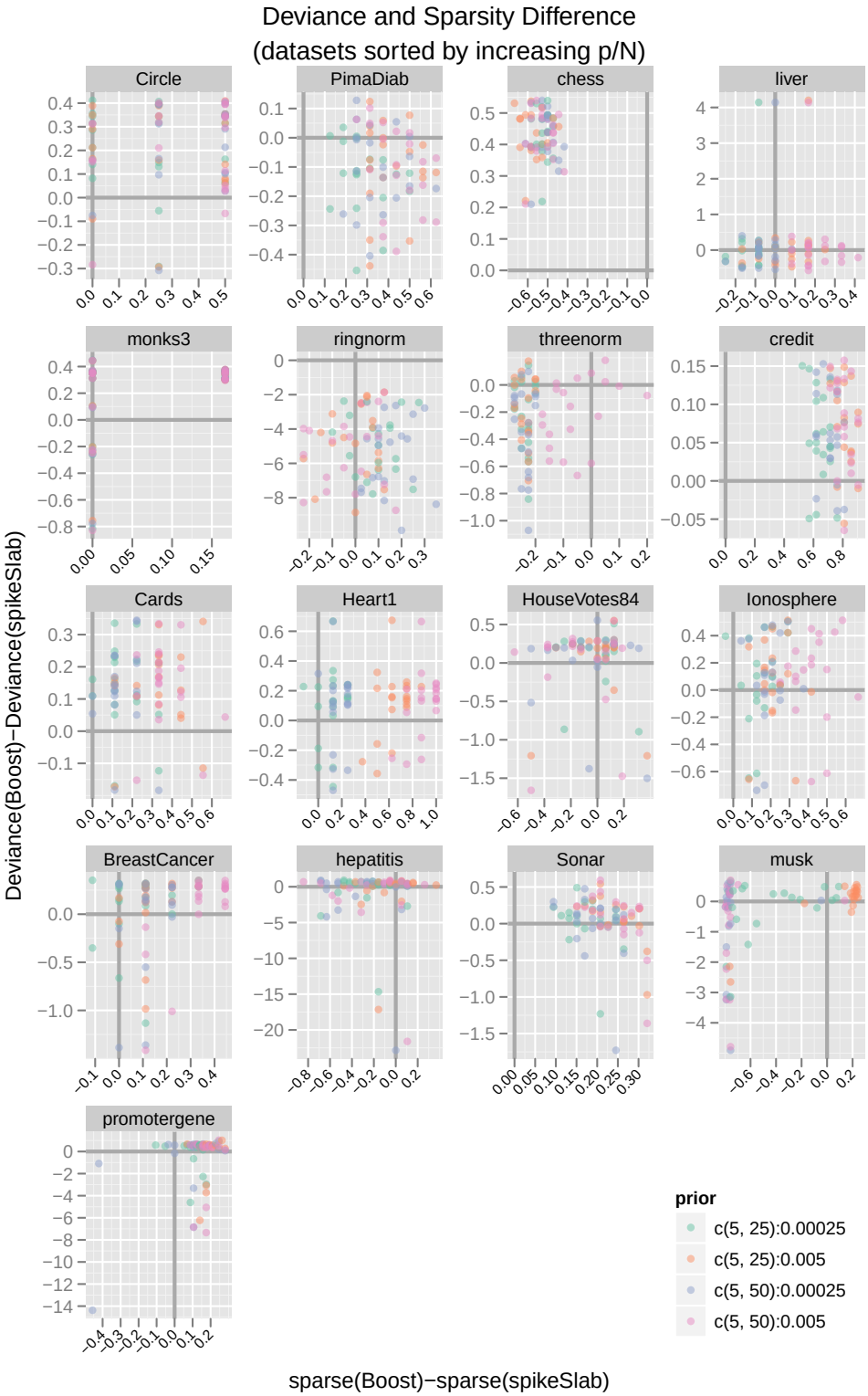

Figure 24: UCI data I: Difference in proportion of included model terms versus differences in predictive deviances.
POsitive values denote smaller deviances/models for our approach compared to mboost. `Spirals, tictactoe, twonorm, titanic` not shown because there were no differences in sparsity.

use slightly different priors and algorithm settings for this model building strategy because the settings suitable for the previous model-building strategy perform less well for very high-dimensional models with $n \lessapprox p$ such as the ones considered here. Specifically, we choose a more informative prior for w to enforce selection of terms. Our results show that using the default uniform prior $w \sim \text{Beta}(1,1)$ tends to yield large models which included almost all possible terms most of the time in these strongly over-parameterized models. We also use NMIG instead of peNMIG for penalization groups with $d = 1$ (i.e. linear terms and binary factors) to reduce the posterior's dimensionality. Reported results are for combinations of $v_0 = 0.005, 0.00025$ and $(a_w, b_w) = (1,1), (20,40)$ with $(a_\tau, b_\tau) = (5,25)$.

Figure 25 shows deviance values for the test data (top) and differences in deviances and sparsity (bottom) between our approach and componentwise boosting with `mboost`. Compared to the results for the models with preselection, predictive performance is worse for these high-dimensional logistic additive models and compares less favorably with results from boosting: With the exception of `credit`, all median deviances are larger than those for `mboost`. The bottom graph shows that our approach deals comparatively less well with high-dimensional, sparse settings (e.g. `musk` with $n = 476$ and 332 potential model terms, of which 166 are smooth terms.) Using an informative prior for w to enforce model sparsity seems to work well for large p/N and does not influence prediction quality in either direction. In settings with smaller p/N the value of v_0 has more influence on the sparsity of the estimated model than (a_w, b_w): Compare the results for `credit`, `Cards`, and `Heart1` (smaller p/N), where the more parsimonious models are those with $v_0 = 0.005$, with the results for the other datasets, where the more parsimonious models are those with $(a_w, b_w) = (20,40)$.

More generally, the performance of peNMIG on the binary classification datasets we used show that it is very competitive to componentwise boosting for a majority of the problems but that relative performance seems to decrease somewhat for very high-dimensional problems with many smooth terms.

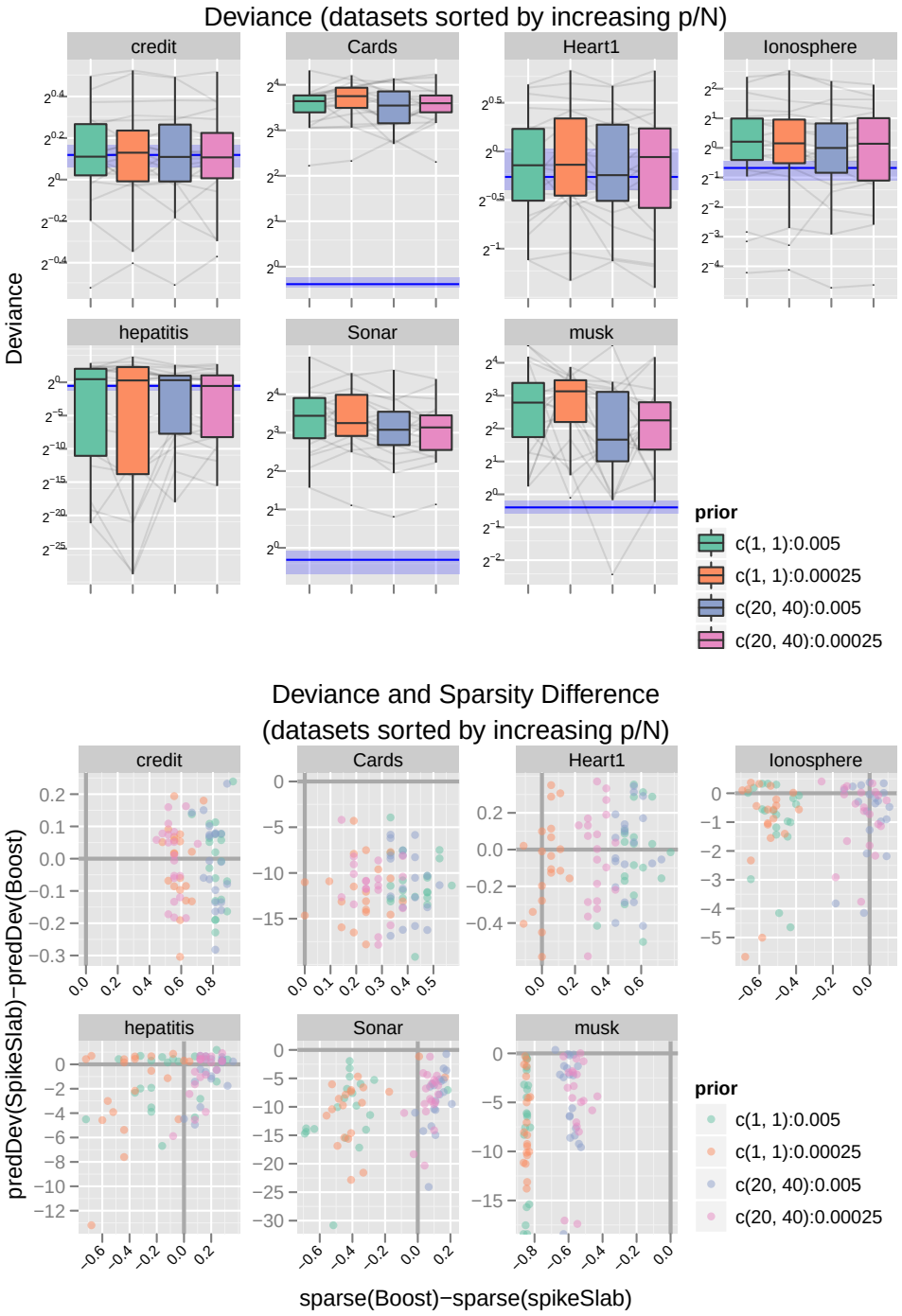

Figure 25: UCI data II:
Upper graph: Predictive deviances for 20-fold crossvalidation. Boxplots show results for the different prior settings, blue symbols show results for mboost: shaded region gives IQR, line represents median. Dark grey lines connect results for the same fold.
Lower graph: Difference in proportion of included model terms versus difference in predictive deviance. Points in topright quadrant denote folds and prior settings in which our approach achieved smaller deviances with a smaller model. Points in lower 2 quadrants denote folds/priors in which our approach resulted in larger deviances than the corresponding mboost-fits.

Dataset	(a_τ, b_τ)	v_0	median($\bar{D}$)		median(AUC)	
			mboost	spikeSlabGAM	mboost	spikeSlabGAM
Circle	(5, 25)	0.00025	0.41	0.11	1.00	1.00
	(5, 25)	0.005	0.41	0.12	1.00	1.00
	(5, 50)	0.00025	0.41	0.12	1.00	1.00
	(5, 50)	0.005	0.41	0.12	1.00	1.00
Spirals	(5, 25)	0.00025	0.98	2.15	0.94	0.91
	(5, 25)	0.005	0.98	2.09	0.94	0.90
	(5, 50)	0.00025	0.98	2.12	0.94	0.91
	(5, 50)	0.005	0.98	2.19	0.94	0.90
titanic	(5, 25)	0.00025	1.09	1.01	0.68	0.68
	(5, 25)	0.005	1.09	1.01	0.68	0.68
	(5, 50)	0.00025	1.09	1.01	0.68	0.68
	(5, 50)	0.005	1.09	1.01	0.68	0.68
PimaDiab	(5, 25)	0.00025	1.03	1.16	0.84	0.81
	(5, 25)	0.005	1.03	1.12	0.84	0.82
	(5, 50)	0.00025	1.03	1.16	0.84	0.80
	(5, 50)	0.005	1.03	1.11	0.84	0.81
chess	(5, 25)	0.00025	0.60	0.15	0.99	1.00
	(5, 25)	0.005	0.60	0.16	0.99	1.00
	(5, 50)	0.00025	0.60	0.15	0.99	1.00
	(5, 50)	0.005	0.60	0.16	0.99	1.00
liver	(5, 25)	0.00025	1.18	1.21	0.79	0.76
	(5, 25)	0.005	1.18	1.22	0.79	0.76
	(5, 50)	0.00025	1.18	1.25	0.79	0.75
	(5, 50)	0.005	1.18	1.23	0.79	0.76
monks3	(5, 25)	0.00025	0.42	0.08	1.00	1.00
	(5, 25)	0.005	0.42	0.08	1.00	1.00
	(5, 50)	0.00025	0.42	0.08	1.00	1.00
	(5, 50)	0.005	0.42	0.09	1.00	1.00
ringnorm	(5, 25)	0.00025	0.89	5.50	0.98	0.99
	(5, 25)	0.005	0.89	5.56	0.98	0.99
	(5, 50)	0.00025	0.89	5.75	0.98	0.99
	(5, 50)	0.005	0.89	5.52	0.98	0.99
threenorm	(5, 25)	0.00025	0.86	1.06	0.93	0.92
	(5, 25)	0.005	0.86	1.05	0.93	0.93
	(5, 50)	0.00025	0.86	1.09	0.93	0.92
	(5, 50)	0.005	0.86	1.08	0.93	0.92
tictactoe	(5, 25)	0.00025	1.03	0.10	0.90	1.00
	(5, 25)	0.005	1.03	0.10	0.90	1.00
	(5, 50)	0.00025	1.03	0.09	0.90	1.00
	(5, 50)	0.005	1.03	0.10	0.90	1.00
twonorm	(5, 25)	0.00025	0.33	4.16	1.00	1.00
	(5, 25)	0.005	0.33	4.12	1.00	1.00
	(5, 50)	0.00025	0.33	4.57	1.00	1.00
	(5, 50)	0.005	0.33	4.54	1.00	1.00
credit	(5, 25)	0.00025	1.11	1.05	0.76	0.75
	(5, 25)	0.005	1.11	1.02	0.76	0.76
	(5, 50)	0.00025	1.11	1.05	0.76	0.75
	(5, 50)	0.005	1.11	1.03	0.76	0.77
Cards	(5, 25)	0.00025	0.83	0.69	0.92	0.93
	(5, 25)	0.005	0.83	0.68	0.92	0.93
	(5, 50)	0.00025	0.83	0.68	0.92	0.93
	(5, 50)	0.005	0.83	0.68	0.92	0.93
Heart1	(5, 25)	0.00025	0.89	0.73	0.94	0.92
	(5, 25)	0.005	0.89	0.73	0.94	0.93
	(5, 50)	0.00025	0.89	0.74	0.94	0.93
	(5, 50)	0.005	0.89	0.71	0.94	0.93
HouseVotes84	(5, 25)	0.00025	0.29	0.04	1.00	1.00
	(5, 25)	0.005	0.29	0.04	1.00	1.00
	(5, 50)	0.00025	0.29	0.03	1.00	1.00
	(5, 50)	0.005	0.29	0.04	1.00	1.00
Ionosphere	(5, 25)	0.00025	0.92	0.80	0.90	0.87
	(5, 25)	0.005	0.92	0.84	0.90	0.88
	(5, 50)	0.00025	0.92	0.86	0.90	0.89
	(5, 50)	0.005	0.92	0.81	0.90	0.88
BreastCancer	(5, 25)	0.00025	0.37	0.13	1.00	1.00
	(5, 25)	0.005	0.37	0.14	1.00	1.00
	(5, 50)	0.00025	0.37	0.13	1.00	1.00
	(5, 50)	0.005	0.37	0.15	1.00	1.00
hepatitis	(5, 25)	0.00025	0.75	0.20	1.00	1.00
	(5, 25)	0.005	0.75	0.21	1.00	1.00
	(5, 50)	0.00025	0.75	0.10	1.00	1.00
	(5, 50)	0.005	0.75	0.09	1.00	1.00
Sonar	(5, 25)	0.00025	1.15	1.03	0.85	0.85
	(5, 25)	0.005	1.15	0.94	0.85	0.86
	(5, 50)	0.00025	1.15	1.03	0.85	0.85
	(5, 50)	0.005	1.15	0.95	0.85	0.86
musk	(5, 25)	0.00025	0.91	0.83	0.91	0.92
	(5, 25)	0.005	0.91	0.74	0.91	0.92
	(5, 50)	0.00025	0.91	1.02	0.91	0.92
	(5, 50)	0.005	0.91	1.02	0.91	0.92
promotergene	(5, 25)	0.00025	0.58	0.00	1.00	1.00
	(5, 25)	0.005	0.58	0.00	1.00	1.00
	(5, 50)	0.00025	0.58	0.00	1.00	1.00
	(5, 50)	0.005	0.58	0.00	1.00	1.00

Table 2: Median deviances and AUCs for test samples of UCI data (Models with preselection).

Dataset	(a_w, b_w)	v_0	median($\bar{D}$)		median(AUC)	
			mboost	spikeSlabGAM	mboost	spikeSlabGAM
`credit`	(1, 1)	0.005	1.09	1.08	0.78	0.80
	(1, 1)	0.00025	1.09	1.09	0.78	0.76
	(20, 40)	0.005	1.09	1.08	0.78	0.80
	(20, 40)	0.00025	1.09	1.08	0.78	0.77
`Cards`	(1, 1)	0.005	0.77	12.47	0.94	0.93
	(1, 1)	0.00025	0.77	13.46	0.94	0.92
	(20, 40)	0.005	0.77	11.67	0.94	0.93
	(20, 40)	0.00025	0.77	12.09	0.94	0.92
`Heart1`	(1, 1)	0.005	0.83	0.91	0.94	0.93
	(1, 1)	0.00025	0.83	0.91	0.94	0.91
	(20, 40)	0.005	0.83	0.84	0.94	0.92
	(20, 40)	0.00025	0.83	0.96	0.94	0.90
`Ionosphere`	(1, 1)	0.005	0.63	1.16	0.98	0.95
	(1, 1)	0.00025	0.63	1.11	0.98	0.95
	(20, 40)	0.005	0.63	1.00	0.98	0.94
	(20, 40)	0.00025	0.63	1.10	0.98	0.95
`hepatitis`	(1, 1)	0.005	0.69	1.49	1.00	1.00
	(1, 1)	0.00025	0.69	1.21	1.00	1.00
	(20, 40)	0.005	0.69	1.25	1.00	1.00
	(20, 40)	0.00025	0.69	0.76	1.00	1.00
`Sonar`	(1, 1)	0.005	0.81	10.86	0.96	0.92
	(1, 1)	0.00025	0.81	9.52	0.96	0.92
	(20, 40)	0.005	0.81	8.44	0.96	0.92
	(20, 40)	0.00025	0.81	8.81	0.96	0.91
`musk`	(1, 1)	0.005	0.76	6.92	0.96	0.92
	(1, 1)	0.00025	0.76	8.77	0.96	0.93
	(20, 40)	0.005	0.76	3.17	0.96	0.91
	(20, 40)	0.00025	0.76	4.79	0.96	0.92

Table 3: Median deviances and AUCs in test samples for UCI data. (Models without preselection)

6.2 Insect Venom Allergy

We reanalyze data on insect venom allergy from a large observational multicenter study previously analyzed in Ruëff et al. [2009]. The data consists of 962 patients from 14 European study centers with established bee or vespid venom allergy who had had an allergic reaction after a field sting. The binary outcome of interest is whether patients suffered a severe, life-threatening reaction, defined as anaphylactic shock, loss of consciousness, or cardiopulmonary arrest, following the index sting. A severe reaction was observed for 206 of the 962 patients (21.4%). Data were collected on

- the concentration of tryptase, a potential biomarker (`logtryp`, $[\log(\mu g/l)]$),
- sex (`sex`),
- age (`age` [years]),
- the culprit insect: bee or vespid (`insect`),
- the intake of cardiovascular medication: β-blockers (`betablocker`), ACE inhibitors (`aceinhibitor`) and/or anti-hypertensive drugs (`heartmeds`),
- whether the patient had had at least one minor systemic reaction to a sting prior to the index sting (`stings`),
- the CAP-class (a measure of antibody load) of the patient with regard to the venom of the culprit insect, with levels $1, 2, 3, 4, \geq 5$ (`cap`).

(R variable names in `brackets`)

An analysis of this data has to take into account possible study center effects, possible non-linear effects of both age and the (logarithm of) blood serum tryptase concentrations and the possibility of differing effect structures for bee and wasp stings. We fit a peNMIG-model with interactions between culprit insect and the other covariates, smooth functions for both age and tryptase and a random intercept for the study center with spikeSlabGAM. The following code example shows the necessary R commands.

```
> formula <- severe ~ insect * (sex + stings + betablocker +
+     aceinhibitor + heartmeds + cap) + sm(age) + sm(logtryp) +
+     rnd(studcent)
> mcmc <- list(chainLength = 10000, burnin = 500, thin = 5,
+     sampleY = TRUE, blocksize = c(10, 15), modeSwitching=FALSE)
> hyper <- list(tau = c(5, 25), gamma = 0.00025)
> m <- spikeAndSlab(formula = formula, data = severity,
+     hyperparameters = hyper, mcmc = mcmc,
+     family = "binomial")

[. . .]
starting...burnin done!
10% 20% 30% 40% 50% 60% 70% 80% 90% 100%
Acceptance: Alpha: 0.683794 Ksi: 0.593714
```

We generate 10000 samples from the posterior, keeping every fifth from a chain with 50000 iterations after a burn in of 1000 iterations. We use hyper-parameters $(a_\tau, b_\tau) = (5, 25)$, $v_0 = 0.00025$ and $a_w = b_w = 1$. We also instruct the sampler to sample $\boldsymbol{y}$ from the posterior predictive distribution for model validation, to use blocksizes of 10 and 15 for the P-IWLS updates of $\boldsymbol{\alpha}$ and $\boldsymbol{\xi}$, respectively, and not to use the mode switching described in Section 4.4. Running the chain takes about 10 minutes a modern desktop PC (Intel Core2 Quad Q9550 CPU with 2.83GHz).

Acceptance rates for both $\boldsymbol{\alpha}$ and $\boldsymbol{\xi}$ are good, and the traceplots (see fig. 26 for the traces of the first nine regression coefficients) indicate that the sampler has converged. Figure 27 shows the estimated smooth terms for age and

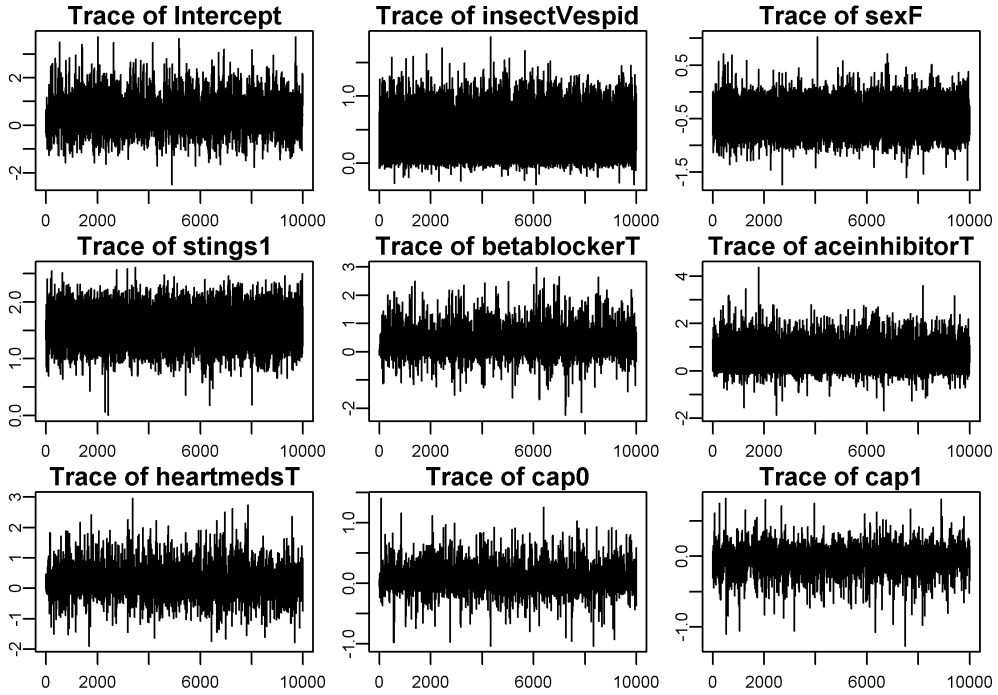

Figure 26: Traceplots for the first 9 entries in $\boldsymbol{\beta}$ for the insect allergy data

the log of tryptase. The estimated function shape for age is very close to a straight line, while the estimated function shape for tryptase is fairly flat for lower values, and than rises more rapidly for larger values.

Inclusion probabilities for the different penalization groups are displayed in table 4. Based on the marginal inclusion probabilities with a cutoff of 0.5, we would select culprit insect, sex, previous stings, ACE inhibitors, both linear and smooth terms for both age and tryptase and the random effect for study center as relevant predictors. This model, which is based on the marginal inclusion probabilities, is also very close to the mode of the posterior of γ. The configuration of γ with the highest posterior probability (p=0.0285) corresponds exactly to this median model, and the configuration of γ with the second highest posterior probability (p=0.0266) is the same but without

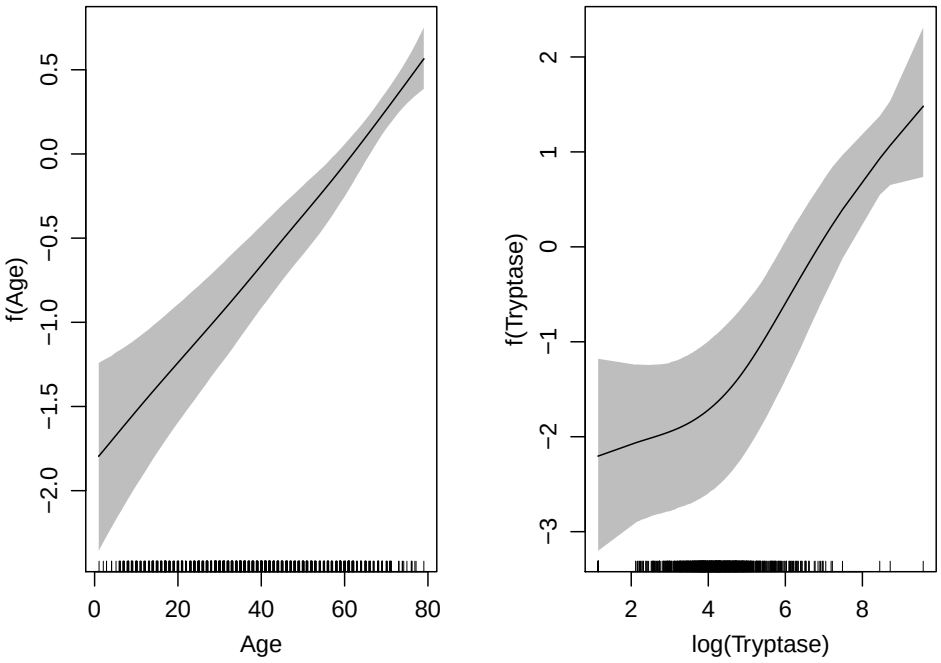

Figure 27: Function estimates and 80% credible regions for the insect allergy data

the smooth term for age, the marginal inclusion probability of which is only marginally above 0.5 and the shape of which, as shown in fig. 27 is not far from a linear shape at all.

Our results replicate the prediction model used by Ruëff et al. [2009], who arrived at the same model minus the smooth term for age via a stepwise selection based on the AIC criterium. The spikeSlabGAM model offers an, albeit small, improvement in prediction accuracy as can be expected from the implicit model averaging - the AUC based on the means from the posterior predictive is 0.79 compared to the 0.731 reported in the original analysis. We report posterior means and credible intervals for the exponentiated coefficients of the included model terms in table 5. The means correspond fairly closely to those reported in the previous analysis, but the credible regions are much wider. This larger variability is caused by the implicit model averaging and is, we believe, a more honest assessment of estimation uncertainty than the confidence intervals derived from a single model resulting from a stepwise selection procedure.

model term	$P(\gamma = 1)$	Inclusion
insect	0.79	x
sex	0.9	x
stings	1	x
betablocker	0.36	
aceinhibitor	0.61	x
heartmeds	0.32	
cap	0.19	
sm.age.fx1	0.99	x
sm.age.s	0.55	x
sm.logtryp.fx1	0.95	x
sm.logtryp.s	0.88	x
rnd.studcent	1	x
insect.sex	0.19	
insect.stings	0.15	
insect.betablocker	0.19	
insect.aceinhibitor	0.21	
insect.heartmeds	0.2	
insect.cap	0.21	

Table 4: Inclusion probabilities for the model terms for the insect allergy data. `sm.age.fx1` denotes the linear term for age, `sm.age.s` the smooth term, analogously for tryptase. `rnd.studcent` is the random intercept for study center and `insect.fnord` denotes the interaction of culprit insect with a covariate fnord.

	mean odds ratio	2.5 %	10 %	90 %	97.5 %
culprit insect: Vespid	1.63	0.97	1.03	2.42	2.99
sex: Female	0.60	0.37	0.44	0.88	1.03
stings: 1+	4.75	2.74	3.35	6.76	8.13
ACE inhibitor: Yes	1.66	0.86	0.97	3.58	6.18
Age	1.03	1.02	1.02	1.04	1.05

Table 5: Posterior means and credible intervals for $\exp(\beta)$ (i.e. the multiplicative effect on the estimated odds for a severe reaction) of the included linear model terms.

7 Conclusion

The focus of this report is on the shrinkage and model selection properties of the novel peNMIG prior in structured additive regression. By introducing this redundant multiplicative parameter expansion combined with a spike-and-slab prior on the level of the hypervariances we are able to select or deselect multiple coefficients (i.e. coefficients for a spline basis or random intercepts associated with a grouping factor) simultaneously in order to guide model choice for generalized additive mixed models.

Extensive simulation studies and application examples show that the performance of the proposed approach is competitive to recently proposed adaptive shrinkage priors and frequentist approaches that address estimation and selection of model terms simultaneously. Estimation performance is very robust against different hyperparameter configurations in all the settings we considered. Variable selection and model choice are more sensitive to varying hyperparameters, but we are confident that the collected simulations and application examples provide a solid foundation for the choice of appropriate values for any analysis.

Our approach is implemented in the R-package spikeSlabGAM. The conditional conjugacy of the proposed prior hierarchy allows for fast and very stable fully Bayesian inference based on MCMC sampling. In its current state, spikeSlabGAM allows fitting additive mixed models for Gaussian, Binomial and Poisson responses. Extensions for geoadditive modeling with GMRFs, multivariate smooth terms and robust error term distributions are straightforward.

Our simulation studies also indicate that peNMIG may be less well suited to very high-dimensional problems. Further research is needed to determine whether this is due to fundamental properties of our proposal such as the doubling of regression coefficients caused by the parameter expansion or whether performance in $p > n$-settings can be redeemed by selecting more appropriate hyperparameters. One promising alternative to P-splines especially for more high dimensional (additive) models and the exploration of interaction effects that we intend to address in future work are low-rank representations of Gaussian process priors.

Acknowledgement

The author wants to thank C.B. Storlie for making the R-code for ACOSSO available. Thomas Kneib and Ludwig Fahrmeir offered valuable feedback on drafts of this report. Ludwig Fahrmeir provided the proof for the pole at zero of the marginal peNMIG prior. We are indebted to Franziska Ruëff for letting us use the insect allergy data set as an application example. Financial support from the German Science Foundation (grant FA 128/5-1) is gratefully acknowledged.

List of Figures

List of Tables

References

A. Asuncion and D.J Newman. *UCI Machine Learning Repository*, 2007. URL `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

K. Bae and B.K. Mallick. Gene selection using a two-level hierarchical Bayesian model. *Bioinformatics*, 20(18):3423–3430, 2004.

M.M. Barbieri and J.O. Berger. Optimal predictive model selection. *Annals of Statistics*, 32(3):870–897, 2004.

Douglas Bates and Martin Maechler. *lme4: Linear mixed-effects models using S4 classes*, 2009. URL `http://CRAN.R-project.org/package=lme4`. R package version 0.999375-33.

A. Brezger and S. Lang. Generalized structured additive regression based on Bayesian P-splines. *Computational Statistics & Data Analysis*, 50(4):967–991, 2006.

B.P. Carlin and S. Chib. Bayesian model choice via Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57 (3):473–484, 1995.

C.M. Carvalho, N.G. Polson, and J.G. Scott. The horseshoe estimator for sparse signals. *Biometrika*, 97(2):465–480, 2010.

R. Cottet, R.J. Kohn, and D.J. Nott. Variable Selection and Model Averaging in Semiparametric Overdispersed Generalized Linear Models. *Journal of the American Statistical Association*, 103(482):661–671, 2008.

P. Dellaportas, J.J. Forster, and I. Ntzoufras. On Bayesian model and variable selection using MCMC. *Statistics and Computing*, 12(1):27–36, 2002.

P.H.C. Eilers and B.D. Marx. Flexible smoothing with B-splines and penalties. *Statistical Science*, 11(2):89–102, 1996.

L. Fahrmeir, T. Kneib, and S. Lang. Penalized structured additive regression for space-time data: a Bayesian perspective. *Statistica Sinica*, 14:731–761, 2004.

L. Fahrmeir, T. Kneib, and S. Konrath. Bayesian regularisation in structured additive regression: a unifying perspective on shrinkage, smoothing and predictor selection. *Statistics and Computing*, 20(2):203–219, 2010.

D. Gamerman. Efficient sampling from the posterior distribution in generalized linear models. *Statistics and Computing*, 7:57–68, 1997.

A. Gelman, D.A. Van Dyk, Z. Huang, and J.W. Boscardin. Using redundant parameterizations to fit hierarchical models. *Journal of Computational and Graphical Statistics*, 17(1):95–122, 2008.

E.I. George and R.E. McCulloch. Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, 88(423):881–889, 1993.

R.B. Gramacy. *monomvn: Estimation for multivariate normal and Student-t data with monotone missingness.*, 2010. URL `http://CRAN.R-project.org/package=monomvn`. R package version 1.8-3.

S. Greven. *Non-Standard Problems in Inference for Additive and Linear Mixed Models*. Cuvillier Verlag, 2007.

F.E. Harrell. *Regression modeling strategies*. Springer New York, 2001.

T. Hastie. Pseudosplines. *Journal of the Royal Statistical Society. Series B*, 58(2): 379–396, 1996.

T. Hothorn, P. Buehlmann, T. Kneib, M. Schmid, and B. Hofner. *mboost: Model-Based Boosting*, 2010. R package version 2.0-0.

H. Ishwaran and J.S. Rao. Spike and slab variable selection: frequentist and Bayesian strategies. *The Annals of Statistics*, 33(2):730–773, 2005.

S. Jackman. *Bayesian analysis for the social sciences*. Wiley, 2009.

T. Kneib. *Mixed model based inference in structured additive regression*. Dr. Hut Verlag, 2006. URL `http://edoc.ub.uni-muenchen.de/archive/00005011/`.

T. Kneib, S. Konrath, and L. Fahrmeir. High-dimensional structured additive regression models: Bayesian regularisation, smoothing and predictive performance. *Applied Statistics*, 2010. *to appear*.

L. Kuo and B. Mallick. Variable selection for regression models. *Sankhyā: The Indian Journal of Statistics, Series B*, 60(1):65–81, 1998.

S. Lang and A. Brezger. Bayesian P-splines. *Journal of Computational and Graphical Statistics*, 13(1):183–212, 2004.

J. Lokhorst, B. Venables, B. Turlach, and M. Maechler. *lasso2: L1 constrained estimation.*, 2009. URL `http://CRAN.R-project.org/package=lasso2`. R package version 1.2-10.

X.L. Meng and D. van Dyk. The EM algorithm–an old folk-song sung to a fast new tune. *Journal of the Royal Statistical Society. Series B*, 59(3):511–567, 1997.

T.J. Mitchell and J.J. Beauchamp. Bayesian variable selection in linear regression. *Journal of the American Statistical Association*, 83(404):1023–1032, 1988.

R.B. O'Hara and M.J. Sillanpää. A Review of Bayesian Variable Selection Methods: What, How, and Which? *Bayesian Analysis*, 4(1):85–118, 2009.

A. Panagiotelis and M. Smith. Bayesian identification, selection and estimation of semiparametric functions in high-dimensional additive models. *Journal of Econometrics*, 143(2):291–316, 2008.

T. Park and G. Casella. The bayesian lasso. *Journal of the American Statistical Association*, 103(482):681–686, 2008.

N.G. Polson and J.G. Scott. Shrink globally, act locally: Sparse bayesian regularization and prediction. In J.M. Bernardo, M.J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith, and M. West, editors, *Bayesian Statistics 9*. Oxford University Press, 2010.

B.J. Reich, C.B. Storlie, and H.D. Bondell. Variable selection in Bayesian smoothing spline ANOVA models: Application to deterministic computer codes. *Technometrics*, 51(2):110, 2009.

F. Ruëff, B. Przybilla, M.B. Biló, U. Müller, F. Scheipl, W. Aberer, J. Birnbaum, A. Bodzenta-Lukaszyk, F. Bonifazi, C. Bucher, et al. Predictors of severe systemic anaphylactic reactions in patients with Hymenoptera venom allergy: Importance of baseline serum tryptase–a study of the European Academy of Allergology and Clinical Immunology Interest Group on Insect Venom Hypersensitivity. *Journal of Allergy and Clinical Immunology*, 124(5):1047–1054, 2009.

F. Scheipl. *RLRsim: Exact (Restricted) Likelihood Ratio tests for mixed and additive models.*, 2010a. URL `http://CRAN.R-project.org/package=RLRsim`. R package version 2.0-4.

F. Scheipl. *amer: Additive mixed models with lme4*, 2010b. URL `http://CRAN.R-project.org/package=amer`. R package version 0.6.6.

F. Scheipl. *spikeSlabGAM: Bayesian model selection for Generalized Additive Mixed Models*, 2010c. R package version 0.3-12.

F. Scheipl, S. Greven, and H. Küchenhoff. Size and power of tests for a zero random effect variance or polynomial regression in additive and linear mixed models. *Computational Statistics & Data Analysis*, 52(7):3283–3299, 2008.

C.B. Storlie, H.D. Bondell, B.J. Reich, and H.H. Zhang. Surface estimation, variable selection, and the nonparametric oracle property. *Statistica Sinica*, 2009. *to appear*.

G. Wahba, Y. Wang, C. Gu, R. Klein, and B. Klein. Smoothing spline ANOVA for exponential families, with application to the Wisconsin Epidemiological Study of Diabetic Retinopathy. *The Annals of Statistics*, 23(6):1865–1895, 1995.

S. Wood, R. Kohn, T. Shively, and W. Jiang. Model selection in spline nonparametric regression. *JRSS-B*, 64(1):119–139, 2002.

S.N. Wood. Thin-plate regression splines. *JRSS-B Statistical Methodology*, 65 (1):95–114, 2003.

S.N. Wood. Fast stable direct fitting and smoothness selection for generalized additive models. *JRSS-B*, 70(3):495, 2008.

P. Yau, R. Kohn, and S. Wood. Bayesian variable selection and model averaging in high-dimensional multinomial nonparametric regression. *Journal of Computational and Graphical Statistics*, 12(1):23–54, 2003.

H.H. Zhang and Y. Lin. Component Selection and Smoothing in Smoothing Spline Analysis of Variance Models. *The Annals of Statistics*, 34:2272–2297, 2003.