LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

INSTITUT FÜR STATISTIK

Gerhard Tutz & Andreas Groll

Variable Selection for Generalized Linear Mixed Models by L_1-Penalized Estimation

Technical Report Number 108, 2011
Department of Statistics
University of Munich

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

Variable Selection for Generalized Linear Mixed Models by L_1-Penalized Estimation

Andreas Groll * Gerhard Tutz †

June 3, 2011

Abstract

Generalized linear mixed models are a widely used tool for modeling longitudinal data. However, their use is typically restricted to few covariates, because the presence of many predictors yields unstable estimates. The presented approach to the fitting of generalized linear mixed models includes an L_1-penalty term that enforces variable selection and shrinkage simultaneously. A gradient ascent algorithm is proposed that allows to maximize the penalized log-likelihood yielding models with reduced complexity. In contrast to common procedures it can be used in high-dimensional settings where a large number of potentially influential explanatory variables is available. The method is investigated in simulation studies and illustrated by use of real data sets.

Keywords: Generalized linear mixed model, Lasso, Gradient ascent, Penalty, Linear models, Variable selection

*Department of Statistics, University of Munich, Akademiestrasse 1, D-80799, Munich, Germany, *email*: andreas.groll@stat.uni-muenchen.de

†Department of Statistics, University of Munich, Akademiestrasse 1, D-80799, Munich, Germany, *email*: tutz@stat.uni-muenchen.de

1 Introduction

Generalized linear mixed models (GLMMs) are widely used to model correlated and clustered responses. Various estimation methods have been proposed ranging from numerical integration techniques (for example Booth and Hobert, 1999) over "joint maximization methods" (Breslow and Clayton, 1993; Schall, 1991), in which parameters and random effects are estimated simultaneously, to fully Bayesian approaches (Fahrmeir and Lang, 1999). Overviews on current methods are found in McCulloch and Searle (2001). Due to the heavy computational problems in GLMMs modeling usually is restricted to few predictor variables. When many predictors are available, estimates become very unstable. Therefore, procedures to select the relevant variables are important in modelling. Classical approaches to the selection of predictors are based on test statistics with the usual stability problems of forward-backward algorithms, which are due to the inherent discreteness of the method (for example Breiman, 1996).

A more timely approach to variable selection is based on boosting methods, which have originally been developed within the machine learning community as a method to improve classification. A first breakthrough was the AdaBoost algorithm proposed by Freund and Schapire (1996). Breiman (1998) considered the AdaBoost algorithm as a gradient descent optimization technique and Friedman (2001) extended boosting methods to include regression problems. Bühlmann and Yu (2003) showed how to fit smoothing splines by boosting base learners and introduced the concept of componentwise boosting, which may be exploited to select predictors. For a detailed overview of componentwise boosting, see Bühlmann and Yu (2003) and Bühlmann and Hothorn (2007). For linear mixed models the incorporation of random effects has been considered by Tutz and Reithinger (2007), first attempts to fit univariate GLMMs were proposed by Tutz and Groll (2010).

An alternative approach to variable selection that has received much attention is based on penalized regression techniques. The Lasso proposed by Tibshirani (1996) has become a very popular approach to regression that uses an L_1-penalty on the regression coefficients. This has the effect that all coefficients are shrunken towards zero and some are set exactly to zero. The basic idea is to maximize the log-likelihood $l(\boldsymbol{\beta})$ of the model while constraining the L_1-norm of the parameter vector $\boldsymbol{\beta}$. Thus one obtains the Lasso estimate

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmax}}\ l(\boldsymbol{\beta}), \qquad \text{subject to } ||\boldsymbol{\beta}||_1 \leq s, \tag{1}$$

with $s \geq 0$ and with $||\cdot||_1$ denoting the L_1-norm. Equivalently the Lasso estimate $\hat{\boldsymbol{\beta}}$ can be

derived by solving the optimization problem

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmax}} \ \left[l(\boldsymbol{\beta}) - \lambda ||\boldsymbol{\beta}||_1 \right], \tag{2}$$

with $\lambda \geq 0$. Both s and λ are tuning parameters that have to be determined, for example by cross-validation. This can be very time-consuming, especially in high-dimensional data settings. Thus, to get computation time under control, in general problems that involve a complex log-likelihood, efficient algorithms are needed to derive the solutions of equations (1) or (2).

For *linear* models the optimization problem of the Lasso can be solved by quadratic programming (Tibshirani, 1996), whereas Osborne et al. (2000) recommend an algorithm considering simultaneously the primal problem and its dual, which is highly efficient and is also applicable in high-dimensional cases. A substantial progress was achieved by the LARS algorithm (Efron et al., 2004), which simultaneously produces the set of Lasso fits for all values of the tuning parameters by following the exact, piecewise linear solution path of $\boldsymbol{\beta}$ as a function of s or λ, respectively, and also inspired the regularization path algorithm for the support vector machine (Hastie et al., 2004). In the last decade several improvements have been designed for the Lasso, e.g. the adaptive Lasso (Zou and Hastie, 2006), SCAD (Fan and Li, 2001), the Elastic Net (Zou and Hastie, 2005), the Dantzig selector (Candes and Tao, 2007), the Double Dantzig (James and Radchenko, 2009) and the VISA (Radchenko and James, 2008).

The Lasso has been extended to more general models, for example Tibshirani (1997) proposed a new method to perform variable selection in the Cox model. He minimizes the partial log-likelihood subject to the L_1-norm of the parameters being bounded by a constant, which is done by an iterative two-step estimation scheme, using alternately reweighted least squares and adaption to the constraint through a quadratic programming procedure. This procedure was improved by Gui and Li (2005), who suggested an iteratively reweighted estimation approach based on the LARS algorithm, called the LARS-Cox procedure. But according to Segal (2006) and Goeman (2010) both algorithms are computational so demanding, that they cannot be used very well in high-dimensional scenarios.

For generalized linear models a flexible and efficient approach is the L_1-regularized path following algorithm by Park and Hastie (2007), who extended the concept of the LARS algorithm (Efron et al., 2004) to generalized linear models. The exact solution coefficients $\hat{\beta}_j$ are computed at particular values of the smoothing parameter λ and then the coefficients are connected in a piecewise linear manner. Another promising approach uses the componentwise gradients, initiating from a starting value $\boldsymbol{\beta}^{(0)}$ and then running through the single coordinates of $\boldsymbol{\beta}$, updating them accordant to the gradient of the penalized likelihood (see e.g. Shevade

and Keerthi, 2003, Kim and Kim, 2004 or Genkin et al., 2007). Recently Goeman (2010) presented another approach based on a combination of gradient ascent optimization with the Newton-Raphson algorithm.

The use of penalization techniques for the selection of variables in mixed models is still in the beginning. For Gaussian mixed models Ni et al. (2010) proposed SCAD penalty techniques. Bondell et al. (2010) considered the iterative case of joint selection for fixed and random effects in linear models. In the following we develop L_1-penalty approaches for the generalized linear mixed model. The method works by combining gradient ascent optimization with the Fisher scoring algorithm and is based on the approach of Goeman (2010). The article is structured as follows. In Section 2 we introduce the GLMM. In Section 3 we present the gradient ascent algorithm with its computational details and give further information about starting values and computation of tuning parameters. Then the performance of the gradient ascent algorithm is investigated in two simulation studies. Applications are considered in Section 4.

2 Generalized Linear Mixed Models - GLMMs

Let y_{it} denote observation t in cluster i, $i = 1, \ldots, n$, $t = 1, \ldots, T_i$, collected in $\mathbf{y}_i^T = (y_{i1}, \ldots, y_{iT_i})$. Let $\mathbf{x}_{it}^T = (1, x_{it1}, \ldots, x_{itp})$ be the covariate vector associated with fixed effects and $\mathbf{z}_{it}^T = (z_{it1}, \ldots, z_{itq})$ be the covariate vector associated with random effects. It is assumed that the observations y_{it} are conditionally independent with means $\mu_{it} = E(y_{it}|\mathbf{b}_i, \mathbf{x}_{it}, \mathbf{z}_{it})$ and variances $var(y_{it}|\mathbf{b}_i) = \phi\upsilon(\mu_{it})$, where $\upsilon(.)$ is a known variance function and ϕ is a scale parameter. The GLMM that we consider in the following has the form

$$g(\mu_{it}) = \mathbf{x}_{it}^T\boldsymbol{\beta} + \mathbf{z}_{it}^T\mathbf{b}_i = \eta_{it}^{\text{par}} + \eta_{it}^{\text{rand}}, \tag{3}$$

where g is a monotonic and continuously differentiable link function, $\eta_{it}^{\text{par}} = \mathbf{x}_{it}^T\boldsymbol{\beta}$ is a linear parametric term with parameter vector $\boldsymbol{\beta}^T = (\beta_0, \beta_1, \ldots, \beta_p)$ including intercept and $\eta_{it}^{\text{rand}} = \mathbf{z}_{it}^T\mathbf{b}_i$ contains the cluster-specific random effects $\mathbf{b}_i \sim N(\mathbf{0}, \mathbf{Q})$, with $q \times q$ covariance matrix $\mathbf{Q}$. An alternative form that we also use is

$$\mu_{it} = h(\eta_{it}), \quad \eta_{it} = \beta_0 + \eta_{it}^{\text{par}} + \eta_{it}^{\text{rand}},$$

where $h = g^{-1}$ is the inverse link function.

A closed representation of model (3) is obtained by using matrix notation. By collecting observations within one cluster, the model has the form

$$g(\boldsymbol{\mu}_i) = \mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{b}_i,$$

where $\mathbf{X}_i^T = (\mathbf{x}_{i1}, \ldots, \mathbf{x}_{iT_i})$ denotes the design matrix of the i-th cluster and $\mathbf{Z}_i^T = (\mathbf{z}_{i1}, \ldots, \mathbf{z}_{iT_i})$. For all observations one obtains

$$g(\boldsymbol{\mu}) = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{b},$$

with $\mathbf{X}^T = [\mathbf{X}_1^T, \ldots, \mathbf{X}_n^T]$ and block-diagonal matrix $\mathbf{Z} = \mathit{Blockdiag}(\mathbf{Z}_1, \ldots, \mathbf{Z}_n)$. For the random effects vector $\mathbf{b}^T = (\mathbf{b}_1^T, \ldots, \mathbf{b}_n^T)$ one has a normal distribution with block-diagonal covariance matrix $\mathbf{Q}_\mathbf{b} = \mathit{diag}(\mathbf{Q}, \ldots, \mathbf{Q})$.

Focusing on GLMMs we assume that the conditional density of y_{it}, given explanatory variables and the random effect $\mathbf{b}_i$, is of exponential family type

$$f(y_{it}|\mathbf{x}_{it}, \mathbf{b}_i) = \exp\left\{\frac{(y_{it}\theta_{it} - \kappa(\theta_{it}))}{\phi} + c(y_{it}, \phi)\right\},$$

where $\theta_{it} = \theta(\mu_{it})$ denotes the natural parameter, $\kappa(\theta_{it})$ is a specific function corresponding to the type of exponential family, $c(.)$ the log normalization constant and ϕ the dispersion parameter (compare Fahrmeir and Tutz, 2001).

One popular method to maximize GLMMs is penalized quasi-likelihood (PQL), which has been suggested by Breslow and Clayton (1993), Lin and Breslow (1996) and Breslow and Lin (1995). Typically the covariance matrix $\mathbf{Q}(\boldsymbol{\varrho})$ of the random effects $\mathbf{b}_i$ depends on an unknown parameter vector $\boldsymbol{\varrho}$. In penalization-based concepts the joint likelihood-function is specified by the parameter vector of the covariance structure $\boldsymbol{\varrho}$ together with the dispersion parameter ϕ, which are collected in $\boldsymbol{\gamma}^T = (\phi, \boldsymbol{\varrho}^T)$, and parameter vector $\boldsymbol{\delta}^T = (\boldsymbol{\beta}^T, \mathbf{b}^T)$. The corresponding log-likelihood is

$$l(\boldsymbol{\delta}, \boldsymbol{\gamma}) = \sum_{i=1}^{n} \log\left(\int f(\mathbf{y}_i|\boldsymbol{\delta}, \boldsymbol{\gamma}) p(\mathbf{b}_i, \boldsymbol{\gamma}) d\mathbf{b}_i\right), \tag{4}$$

where $p(\mathbf{b}_i, \boldsymbol{\gamma})$ denotes the density of the random effects. Breslow and Clayton (1993) derived the approximation

$$l^{\text{app}}(\boldsymbol{\delta}, \boldsymbol{\gamma}) = \sum_{i=1}^{n} \log(f(\mathbf{y}_i|\boldsymbol{\delta}, \boldsymbol{\gamma})) - \frac{1}{2}\mathbf{b}^T\mathbf{Q}(\boldsymbol{\varrho})^{-1}\mathbf{b}, \tag{5}$$

where the penalty term $\mathbf{b}^T\mathbf{Q}(\boldsymbol{\varrho})^{-1}\mathbf{b}$ is due to the approximation based on the Laplace method.

PQL usually works within the profile likelihood concept. It is distinguished between the estimation of $\boldsymbol{\delta}$, given the plugged-in estimate $\hat{\boldsymbol{\gamma}}$, resulting in the profile-likelihood $l^{\text{app}}(\boldsymbol{\delta}, \hat{\boldsymbol{\gamma}})$, and the estimation of $\boldsymbol{\gamma}$. The PQL method is implemented in the macro `GLIMMIX` and proc `GLMMIX` in SAS (Wolfinger, 1994), in the `glmmPQL` and `gamm` functions of the R-packages `MASS` (Venables and Ripley, 2002) and `mgcv` (Wood, 2006). Further notes were given by Wolfinger and O'Connell (1993), Littell et al. (1996) and Vonesh (1996).

3 Regularization in GLMMs

In the following the log-likelihood (4) is expanded to include the penalty term $\lambda \sum_{i=1}^{p} |\beta_i|$. Approximation along the lines of Breslow and Clayton (1993) yields the penalized log-likelihood

$$l^{\text{pen}}(\boldsymbol{\beta}, \mathbf{b}, \boldsymbol{\gamma}) = l^{\text{pen}}(\boldsymbol{\delta}, \boldsymbol{\gamma}) = l^{\text{app}}(\boldsymbol{\delta}, \boldsymbol{\gamma}) - \lambda \sum_{i=1}^{p} |\beta_i|. \tag{6}$$

For given $\hat{\boldsymbol{\gamma}}$ the optimization problem reduces to

$$\hat{\boldsymbol{\delta}} = \underset{\boldsymbol{\delta}}{\text{argmax}}\ l^{\text{pen}}(\boldsymbol{\delta}, \hat{\boldsymbol{\gamma}}) = \underset{\boldsymbol{\delta}}{\text{argmax}} \left[l^{\text{app}}(\boldsymbol{\delta}, \hat{\boldsymbol{\gamma}}) - \lambda \sum_{i=1}^{p} |\beta_i| \right]. \tag{7}$$

We will use a full gradient algorithm that is based on the algorithm of Goeman (2010). As Goeman (2010) already pointed out, the algorithm can easily be amended to situations in which some parameters should not be penalized. In this case the penalty term from the optimization problem of equation (2) is replaced by $\sum_{i=1}^{p} \lambda_i |\beta_i|$, where $\lambda_i = 0$ is chosen for unpenalized parameters. The penalty used in (6) and (7) can be seen as a partially penalized approach if the whole parameter vector $\boldsymbol{\delta}^T = (\boldsymbol{\beta}^T, \mathbf{b}^T)$ is considered.

3.1 Gradient Ascent Algorithm - glmmLasso

In the following an algorithm is presented for maximizing the penalized log-likelihood $l^{\text{pen}}(\boldsymbol{\delta}, \boldsymbol{\gamma})$ from equation (6). In contrast to the approaches of Shevade and Keerthi (2003), Kim and Kim (2004) and Genkin et al. (2007), where only a single component is updated at a time, it follows the gradient of the likelihood from a given starting value of $\boldsymbol{\delta}$ and uses the full gradient at each step. Similar to Goeman (2010) the algorithm can automatically switch to a Fisher scoring procedure when it gets close to the optimum and therefore avoids the tendency to slow convergence which is typical for gradient ascent algorithms. An additional step is needed to estimate the variance-covariance components $\mathbf{Q}$ of the random effects. To keep the notation simple, we omit the argument $\boldsymbol{\gamma}$ in the following description of the algorithm and write $l^{\text{app}}(\boldsymbol{\delta})$ instead of $l^{\text{app}}(\boldsymbol{\delta}, \boldsymbol{\gamma})$.

Algorithm `glmmLasso`

---

1. *Initialization*

   Compute starting values $\hat{\boldsymbol{\beta}}^{(0)}, \hat{\mathbf{b}}^{(0)}, \hat{\boldsymbol{\gamma}}^{(0)}$ (see Section 3.2.1) and set $\hat{\boldsymbol{\eta}}^{(0)} = \mathbf{X}\hat{\boldsymbol{\beta}}^{(0)} + \mathbf{Z}\hat{\mathbf{b}}^{(0)}$.

2. *Iteration*

For $l = 1, 2, \ldots$ until convergence:

(a) *Calculation of the log-likelihood gradient for given* $\hat{\boldsymbol{\gamma}}^{(l-1)}$

With $\mathbf{s}(\boldsymbol{\delta}) = \partial l^{\text{app}}(\boldsymbol{\delta})/\partial \boldsymbol{\delta}$ derive:

$$s_0^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l-1)}) = s_0(\hat{\boldsymbol{\delta}}^{(l-1)}), \qquad s_i^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l-1)}) = s_i(\hat{\boldsymbol{\delta}}^{(l-1)}), \quad i = p+1, \ldots, p+ns.$$

Furthermore, for $i = 1, \ldots, p$ derive:

$$s_i^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l-1)}) = \begin{cases} s_i(\hat{\boldsymbol{\delta}}^{(l-1)}) - \lambda \operatorname{sign}(\hat{\beta}_i^{(l-1)}) & \text{if } \hat{\beta}_i^{(l-1)} \neq 0 \\ s_i(\hat{\boldsymbol{\delta}}^{(l-1)}) - \lambda \operatorname{sign}(s_i(\hat{\boldsymbol{\delta}}^{(l-1)})) & \text{if } \hat{\beta}_i^{(l-1)} = 0 \text{ and } |s_i(\hat{\boldsymbol{\delta}}^{(l-1)})| > \lambda \\ 0 & \text{otherwise} \end{cases},$$

where

$$\operatorname{sign}(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0 \\ -1 & \text{if } x < 0. \end{cases}$$

(b) *Calculation of the dircetional second derivative*

Let $\mathbf{A} := [\mathbf{X}, \mathbf{Z}]$ and $\mathbf{K} = diag(0, \ldots, 0, \mathbf{Q}^{-1}, \ldots, \mathbf{Q}^{-1})$ be a block-diagonal penalty matrix with a diagonal of $p+1$ zeros corresponding to the fixed effects and then n times the matrix $\mathbf{Q}^{-1}$. Then the Fisher matrix is given in closed form as $\mathbf{F}^{\text{pen}}(\boldsymbol{\delta}) = \mathbf{A}^T\mathbf{W}(\boldsymbol{\delta})\mathbf{A} + \mathbf{K}$, with $\mathbf{W}(\boldsymbol{\delta}) = \mathbf{D}(\boldsymbol{\delta})\boldsymbol{\Sigma}^{-1}(\boldsymbol{\delta})\mathbf{D}(\boldsymbol{\delta})^T$ and $\mathbf{D}(\boldsymbol{\delta}) = \partial h(\boldsymbol{\eta})/\partial \boldsymbol{\eta}, \boldsymbol{\Sigma}(\boldsymbol{\delta}) = cov(\mathbf{y}|\boldsymbol{\delta})$. The directional second derivative is given for every $\boldsymbol{\delta}$ and every direction vector $\mathbf{v} \in \mathbb{R}^{p+1+ns}$ by

$$l''_{\text{pen}}(\boldsymbol{\delta}; \mathbf{v}) = \mathbf{v}^T\mathbf{F}^{\text{pen}}(\boldsymbol{\delta})\mathbf{v}$$

(c) *Optimum of Taylor approximation*

Based on the Taylor approximation used in Goeman (2010), we derive

$$t_{\text{edge}}^{(l-1)} = \min_i \left\{ -\frac{\hat{\delta}_i^{(l-1)}}{s_i^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l-1)})} : \operatorname{sign}(\hat{\delta}_i^{(l-1)}) = -\operatorname{sign}[s_i^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l-1)})] \neq 0 \right\}$$

and

$$t_{\text{opt}}^{(l-1)} = \frac{||\mathbf{s}^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l-1)})||_2}{l''_{\text{app}}(\hat{\boldsymbol{\delta}}^{(l-1)}, \mathbf{s}^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l-1)}))},$$

with $||\cdot||_2$ denoting the L_2 norm.

(d) *Update*

$$\hat{\boldsymbol{\delta}}^{(l)} = \begin{cases} \hat{\boldsymbol{\delta}}^{(l-1)} + t_{\text{edge}}^{(l-1)} \mathbf{s}^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l-1)}) & \text{if } t_{\text{opt}}^{(l-1)} \geq t_{\text{edge}}^{(l-1)} \\ \hat{\boldsymbol{\delta}}_{\text{NR}}^{(l-1)} & \text{if } t_{\text{opt}}^{(l-1)} < t_{\text{edge}}^{(l-1)} \text{ and } \text{sign}(\hat{\boldsymbol{\delta}}_{\text{NR}}^{(l)}) = \text{sign}(\hat{\boldsymbol{\delta}}^{(l-1)}) \\ \hat{\boldsymbol{\delta}}^{(l-1)} + t_{\text{opt}}^{(l-1)} \mathbf{s}^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l-1)}) & \text{otherwise,} \end{cases}$$

where $\hat{\boldsymbol{\delta}}_{\text{NR}}^{(l)}$ denotes the Fisher scoring estimate as given in Section 3.2.2.

(e) *Computation of variance-covariance components*
Estimates $\hat{\mathbf{Q}}^{(l)}$ are obtained as approximate EM-type estimates or by alternative methods (see Section 3.2.3) yielding the update $\boldsymbol{\varrho}^{(l)}$. If necessary, the whole vector $\hat{\boldsymbol{\gamma}}^{(l)}$ is completed by an estimate of the dispersion parameter.

3. *Re-Estimation*
In a final step a model that includes only the variables corresponding to non-zero parameters of $\hat{\boldsymbol{\beta}}$ is fitted. A simple Fisher scoring, resulting in the final estimates $\hat{\boldsymbol{\delta}}, \hat{\mathbf{Q}}$ is used.

3.2 Computational Details of `glmmLasso`

In the following we give a more detailed description of the single steps of the `glmmLasso` algorithm. First details of the computation of starting values are given and then two estimation techniques for the variance-covariance components are described.

3.2.1 Starting Values for `glmmLasso`

We compute the starting values $\hat{\boldsymbol{\beta}}^{(0)}, \hat{\mathbf{b}}^{(0)}, \hat{\mathbf{Q}}^{(0)}$ from step 1 of the `glmmLasso` algorithm by fitting the simple global intercept model with random effects given by, $g(\mu_{it}) = \beta_0 + \mathbf{z}_{it}^T \mathbf{b}_i$. This can be done very easily, for example by using the R-function `glmmPQL` (Wood, 2006) from the `MASS` library (Venables and Ripley, 2002).

3.2.2 Fisher Scoring

Similar to Goeman (2010) we combine gradient ascent optimization with the Fisher scoring algorithm in the update step 2 (d) of the `glmmLasso` algorithm. Although gradient ascent optimization is computationally simple, because no matrix inversion or other computationally expensive calculations are involved, often a large number of steps is required for convergence. By allowing the algorithm to switch to the Fisher scoring algorithm the algorithm becomes much faster.

For an arbitrary iteration we define $J = \{j : \text{sign}(\beta_j) \neq 0, j = 0, 1, \ldots, p\}$, the index set of the "active" covariates, corresponding to the $m = \#J \leq p+1$ non-zero coefficients. Furthermore, let $\tilde{\boldsymbol{\delta}}^T = (\beta_{J_1}, \ldots, \beta_{J_m}, \mathbf{b}^T)$, and let $\tilde{\mathbf{s}}^{\text{pen}}(\boldsymbol{\delta}) = \left\{s^{\text{pen}}_{J_1}(\boldsymbol{\delta}), \ldots, s^{\text{pen}}_{J_m}(\boldsymbol{\delta}), s^{\text{pen}}_{p+1}(\boldsymbol{\delta}), \ldots, s^{\text{pen}}_{p+ns}(\boldsymbol{\delta})\right\}^T$ be the gradient in the constrained domain and $\tilde{\mathbf{F}}$ the $(m+ns) \times (m+ns)$ Fisher matrix of the constrained optimization, given by $\tilde{\mathbf{F}}(\boldsymbol{\delta}) = \mathbf{A}_J^T \mathbf{W}(\boldsymbol{\delta}) \mathbf{A}_J + \mathbf{K}_J$, with $\mathbf{A}_J := [\mathbf{X}_J, \mathbf{Z}]$, whereas $\mathbf{X}_J$ contains only those columns of $\mathbf{X}$ corresponding to J, and block-diagonal penalty matrix $\mathbf{K}_J = diag(0, \ldots, 0, \mathbf{Q}^{-1}, \ldots, \mathbf{Q}^{-1})$ with a diagonal of m zeros corresponding to the non-zero fixed effects and then n times the matrix $\mathbf{Q}^{-1}$.

One step of Fisher scoring in the current subdomain takes the form

$$\hat{\tilde{\boldsymbol{\delta}}}^{(l)} = \hat{\tilde{\boldsymbol{\delta}}}^{(l-1)} + \left(\tilde{\mathbf{F}}(\hat{\boldsymbol{\delta}}^{(l-1)})\right)^{-1} \tilde{\mathbf{s}}^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l-1)}).$$

This estimator can be mapped back to a $(p + 1 + ns)$-vector $\hat{\boldsymbol{\delta}}^{(l)}_{NR}$ by augmenting $\hat{\tilde{\boldsymbol{\delta}}}^{(l)}$ with zeros for all non-active covariates. In order that the Taylor approximation which is underlying such a step of Fisher scoring holds within the current subdomain, $\hat{\boldsymbol{\delta}}^{(l)}_{NR}$ is accepted only when $\text{sign}(\hat{\boldsymbol{\delta}}^{(l)}_{\text{NR}}) = \text{sign}(\hat{\boldsymbol{\delta}}^{(l-1)})$.

As Goeman (2010) pointed out, it is often better to avoid the attempt of trying a Fisher scoring step whenever it is likely to fail, because it can be computational expensive. Practical experience with our `glmmLasso` algorithm has shown the same tendencies. We do not try a Fisher scoring step at $l = 0$ and after a Fisher scoring step has failed we try another step of Fisher scoring not until the active set has changed. Nevertheless the incorporation of Fisher scoring into the procedure can greatly speed up convergence once the algorithm gets close to the optimum.

3.2.3 Variance-Covariance Components

Variance estimates for the random effects can be derived as an approximate EM algorithm, using the posterior mode estimates and posterior curvatures. One derives $(\mathbf{F}^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l)}))^{-1}$, the inverse of the penalized pseudo Fisher matrix, using the posterior mode estimates $\hat{\boldsymbol{\delta}}^{(l)}$ to obtain the posterior curvatures $\hat{\mathbf{V}}^{(l)}_{ii}$. Now compute $\hat{\mathbf{Q}}^{(l)}$ by

$$\hat{\mathbf{Q}}^{(l)} = \frac{1}{n} \sum_{i=1}^{n} (\hat{\mathbf{V}}^{(l)}_{ii} + \hat{\mathbf{b}}^{(l)}_i (\hat{\mathbf{b}}^{(l)}_i)^T). \tag{8}$$

In general, the $\mathbf{V}_{ii}$ are derived via the formula

$$\mathbf{V}_{ii} = \mathbf{F}_{ii}^{-1} + \mathbf{F}_{ii}^{-1} \mathbf{F}_{i\boldsymbol{\beta}} (\mathbf{F}_{\boldsymbol{\beta}\boldsymbol{\beta}} - \sum_{i=1}^{n} \mathbf{F}_{\boldsymbol{\beta} i} \mathbf{F}_{ii}^{-1} \mathbf{F}_{i\boldsymbol{\beta}})^{-1} \mathbf{F}_{\boldsymbol{\beta} i} \mathbf{F}_{ii}^{-1},$$

where $\mathbf{F}_{\boldsymbol{\beta}\boldsymbol{\beta}}, \mathbf{F}_{i\boldsymbol{\beta}}, \mathbf{F}_{ii}$ are elements of the partitioned Fisher matrix, see Appendix A.

For an alternative estimation of variances (Breslow and Clayton, 1993) maximize the profile likelihood that is associated with the normal theory model. By replacing $\boldsymbol{\beta}$ with $\hat{\boldsymbol{\beta}}$ one maximizes

$$\begin{aligned} l(\mathbf{Q_b}) &= -\frac{1}{2}\log(|\mathbf{V}(\hat{\boldsymbol{\delta}})|) - \frac{1}{2}\log(|\mathbf{X}^T\mathbf{V}^{-1}(\hat{\boldsymbol{\delta}})\mathbf{X}|) \\ &\quad -\frac{1}{2}(\tilde{\boldsymbol{\eta}}(\hat{\boldsymbol{\delta}}) - \mathbf{X}\hat{\boldsymbol{\beta}})^T\mathbf{V}^{-1}(\hat{\boldsymbol{\delta}})(\tilde{\boldsymbol{\eta}}(\hat{\boldsymbol{\delta}}) - \mathbf{X}\hat{\boldsymbol{\beta}}) \end{aligned} \tag{9}$$

with respect to $\mathbf{Q_b}$, with the pseudo-observations $\tilde{\boldsymbol{\eta}}(\boldsymbol{\delta}) = \mathbf{A}\boldsymbol{\delta} + \mathbf{D}^{-1}(\boldsymbol{\delta})(\mathbf{y} - \boldsymbol{\mu}(\boldsymbol{\delta}))$ and with matrices $\mathbf{V}(\boldsymbol{\delta}) = \mathbf{W}^{-1}(\boldsymbol{\delta}) + \mathbf{Z}\mathbf{Q_b}\mathbf{Z}^T$, $\mathbf{Q_b} = Blockdiag(\mathbf{Q}, \ldots, \mathbf{Q})$ and $\mathbf{W}(\boldsymbol{\delta}) = \mathbf{D}(\boldsymbol{\delta})\boldsymbol{\Sigma}^{-1}(\boldsymbol{\delta})\mathbf{D}(\boldsymbol{\delta})^T$. Having calculated $\hat{\boldsymbol{\delta}}^{(l)}$ in the l-th iteration, we obtain the estimator $\hat{\mathbf{Q}}_{\mathbf{b}}^{(l)}$, which is an approximate REML-type estimate for $\mathbf{Q_b}$.

3.3 Incorporation of Categorical Predictors

A frequently found type of structured regressors are categorical predictors (factors), which are usually dummy-coded and hence result in groups of dummy variables. That means a one-dimensional variable is transformed into a group of variables. By construction, the standard Lasso solution is only able to select distinct dummy variables but not whole factors. Since one wants variable selection the algorithm has to be modified in the spirit of the group Lasso, which was proposed by Yuan and Lin (2006). It was explicitly designed for the selection of grouped variables in the form of dummy-coded factors in the usual linear regression set-up and represents an elegant combination of penalization within groups of variables and groupwise selection by using a Lasso penalty at the factor level, and a Ridge-type penalization within coefficient groups.

Meier et al. (2008) have extended the group Lasso to logistic regression and present an efficient algorithm to solve the corresponding convex optimization problem. Their resulting logistic group Lasso estimator is obtained by replacing the Lasso penalty term from equation (2) by the penalty $\sum_{g=1}^{G} \lambda_g ||\boldsymbol{\beta}_{I_g}||_2$, where I_g denotes the index set of to the g-th group of variables, $g = 1, \ldots, G$ and $\lambda_g = \lambda\sqrt{\mathrm{df}_g}$, with df_g representing the number of parameters of group g, which is equal to the number of factor levels minus one for categorical predictors and df_g=1 for continuous predictors.

Suppose that the $p+1$ columns of our design matrix $\mathbf{X}$ are now resulting from G predictors, which may be categorical or continuous, plus intercept. Using the same notations as above, we incorporate the penalization adjustment of Meier et al. (2008) into the `glmmLasso` algorithm by simply modifying step 2 (a) in the following way:

(a2) *Calculation of the log-likelihood gradient*

With $\mathbf{s}(\boldsymbol{\delta}) = \partial l^{\text{app}}(\boldsymbol{\delta})/\partial \boldsymbol{\delta}$ derive:

$$s_0^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l-1)}) = s_0(\hat{\boldsymbol{\delta}}^{(l-1)}), \qquad s_i^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l-1)}) = s_i(\hat{\boldsymbol{\delta}}^{(l-1)}), \quad i = p+1, \ldots, p+ns.$$

Furthermore, for $g = 1, \ldots, G$ derive:

$$\mathbf{s}_{I_g}^{\text{pen}}(\hat{\boldsymbol{\delta}}^{(l-1)}) = \begin{cases} \mathbf{s}_{I_g}(\hat{\boldsymbol{\delta}}^{(l-1)}) - \lambda_g \operatorname{sign}(\hat{\boldsymbol{\beta}}_{I_g}^{(l-1)}) & \text{if } ||\hat{\boldsymbol{\beta}}_{I_g}^{(l-1)}||_2 \neq 0 \\ \mathbf{s}_{I_g}(\hat{\boldsymbol{\delta}}^{(l-1)}) - \lambda_g \operatorname{sign}(\mathbf{s}_{I_g}(\hat{\boldsymbol{\delta}}^{(l-1)})) & \text{if } ||\hat{\boldsymbol{\beta}}_{I_g}^{(l-1)}||_2 = 0 \text{ and } ||\mathbf{s}_{I_g}(\hat{\boldsymbol{\delta}}^{(l-1)})||_2 > \lambda_g \\ \mathbf{0} & \text{otherwise.} \end{cases}$$

3.4 Simulation Study

In the following small simulation study the performance of the `glmmLasso` algorithm is compared to alternative approaches.

Poisson Link The underlying model is the random intercept Poisson model

$$\begin{aligned} \eta_{it} &= \sum_{j=1}^{p} x_{itj}\beta_j + b_i, \quad i = 1, \ldots, 40, \quad t = 1, \ldots, 10, \\ E[y_{it}] &= \exp(\eta_{it}) := \lambda_{it}, \qquad y_{it} \sim \text{Pois}(\lambda_{it}), \end{aligned}$$

with linear effects given by $\beta_1 = -4, \beta_2 = -6, \beta_3 = 10$ and $\beta_j = 0, j = 4, \ldots, 50$. We chose the different settings $p = 3, 5, 10, 20, 50$. For $j = 1, \ldots, 50$ the vectors $\mathbf{x}_{it}^T = (x_{it1}, \ldots, x_{it50})$ follow a uniform distribution within the interval $[-0.14, 0.14]$. The number of observations was determined by $n = 40, T_i := T = 10, i = 1, \ldots, n$. The random effect and the noise variable have been specified by $b_i \sim N(0, \sigma_b^2)$ with $\sigma_b^2 = 0.4, 0.8, 1.6$.

The performance of estimators was evaluated separately for the structural components and the variance. We compare the results of our `glmmLasso` algorithm with the results obtained by the R-functions `glmmPQL` (Venables and Ripley, 2002), `glmmML` (Broström, 2009) and `glmer` (Bates and Maechler, 2010). The `glmmPQL` routine is supplied b¡ the `MASS` library. It operates by iteratively calling the R-function `lme` from the `nlme` library and returns the fitted `lme` model object for the working model at convergence. For more details about the `lme` function, see Pinheiro and Bates (2000). The `glmer` function available in the `lme4` package (Bates and Maechler, 2010) features two different methods of approximating the integrals in the log-likelihood function, Laplace and Gauss-Hermite. We focused on the Gauss-Hermite method using 15 quadrature points. In some cases the `glmer` function did not converge (n.c.), see the

corresponding columns in Table 1 and 2.

Another function that is able to fit the underlying model is the `glmmML` function supplied with the `glmmML` package (Broström, 2009). The function also features two different methods of approximating the integrals in the log-likelihood function, Laplace and Gauss-Hermite. For the first method the results coincide with the results of the `glmmPQL` routine, so we focused on the Gauss-Hermite method in our simulations. Also the `glmmML` function had some convergence problems, which is summarized in the "n.c." columns in Table 1 and 2.

Furthermore we compare our results with two boosting functions, `bGLMM` (EM) and `bGLMM` (REML), introduced in Tutz and Groll (2010), which perform variable selection by boosting techniques. They differ in the computation of the covariance matrix components $\mathbf{Q}$ of the random effects. The first one can be derived as an approximate EM algorithm, the second one by maximizing the profile likelihood that is associated with the normal theory model and therefore could be seen as an approximate REML-type estimate.

By averaging across 100 training data sets we consider mean squared errors for $\boldsymbol{\beta}$ and σ_b given by $\text{mse}_{\boldsymbol{\beta}} := ||\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}||^2$, $\text{mse}_{\sigma_b} := ||\sigma_b - \hat{\sigma}_b||^2$. The means of both quantities are presented in Table 1 and 2. The results of $\text{mse}_{\boldsymbol{\beta}}$ are illustrated in Figure 1, which shows boxplots of the ratios $\log(\text{mse}_{\boldsymbol{\beta}}(\cdot)/\text{mse}_{\boldsymbol{\beta}}(\texttt{glmmPQL}))$ for the different methods, for different numbers of noise variables and the scenario $\sigma_b = 0.4$. Additionally, we present boxplots of the ratios $\log(\text{mse}_{\sigma_b}(\cdot)/\text{mse}_{\sigma_b}(\texttt{glmmPQL}))$ corresponding to $\sigma_b = 0.4$ in Figure 4.

Additional information on the performance of the algorithm was collected in *falseneg* (f.n.), the mean over all 100 simulations of the number of variables $\beta_j, j = 1, 2, 3$, that were not selected and in *falsepos* (f.p.), the mean over all 100 simulations of the number of variables $\beta_j, j = 4, \ldots, 50$, that were selected. It should be noted that the three R-functions are not able to perform variable selection and therefore always estimate all p parameters β_j.

The results for varying number p of covariates $x_{it1}, \ldots, x_{itp}$ are summarized in Table 1 and 2. It is seen that Lasso estimates for $\boldsymbol{\beta}$ distinctly outperform the standard R functions when redundant variables are present and are comparable to the boosting results. An advantage of L_1-penalization over boosting techniques is that it also performs well when all variables in the predictor are influential. Also for the variance component σ_b the `glmmLasso` algorithm slightly outperforms both boosting approaches.

Figure 1 compares the performance of the procedures with `glmmPQL` as the reference. It shows the $\log(\text{mse}_{\boldsymbol{\beta}}(\cdot)/\text{mse}_{\boldsymbol{\beta}}(\texttt{glmmPQL}))$ over the simulations.

		glmmPQL	glmmML		glmer		glmmLasso			bGLMM (EM)			bGLMM (REML)		
σ_b	p	$\text{mse}_{\boldsymbol{\beta}}$	$\text{mse}_{\boldsymbol{\beta}}$	n.c.	$\text{mse}_{\boldsymbol{\beta}}$	n.c.	$\text{mse}_{\boldsymbol{\beta}}$	f.p.	f.n.	$\text{mse}_{\boldsymbol{\beta}}$	f.p.	f.n.	$\text{mse}_{\boldsymbol{\beta}}$	f.p.	f.n.
0.4	3	0.909	0.907	0	0.907	0	0.907	0	0	1.694	0	0.01	1.710	0	0
0.4	5	1.399	1.400	0	1.400	0	1.148	0.53	0	1.694	0	0.01	1.710	0	0
0.4	10	2.710	2.707	0	2.706	0	1.291	0.71	0	1.751	0.02	0.01	1.764	0.02	0
0.4	20	5.646	5.644	0	5.643	0	1.500	0.97	0	1.879	0.08	0.01	1.859	0.06	0
0.4	50	17.268	17.221	0	17.220	0	1.949	1.23	0	2.228	0.21	0.01	2.167	0.19	0
0.8	3	0.844	0.844	0	0.844	0	0.843	0	0	0.979	0	0	0.981	0	0
0.8	5	1.348	1.349	0	1.349	0	1.097	0.44	0	1.008	0.01	0	1.009	0.01	0
0.8	10	2.613	2.612	0	2.611	0	1.419	1.07	0	1.123	0.07	0	1.124	0.07	0
0.8	20	5.456	5.445	0	5.444	0	1.785	1.43	0	1.344	0.17	0	1.342	0.17	0
0.8	50	16.209	16.096	0	16.093	0	1.931	2.32	0	1.686	0.33	0	1.679	0.33	0
1.6	3	0.636	0.450	7	0.446	1	0.438	0	0	0.669	0	0	0.605	0	0
1.6	5	0.994	0.718	7	0.707	1	0.564	0.62	0	0.712	0.05	0	0.648	0.05	0
1.6	10	1.451	1.446	7	1.420	1	0.809	2.26	0	0.741	0.07	0	0.677	0.07	0
1.6	20	3.045	3.089	7	3.094	3	1.177	5.11	0	0.823	0.17	0	0.759	0.16	0
1.6	50	11.127	11.328	7	11.247	3	2.961	10.70	0.01	1.098	0.44	0	1.046	0.45	0

Table 1: Mean squared errors for $\boldsymbol{\beta}$ for the `glmmLasso` and alternative approaches on Poisson data

		glmmPQL	glmmML		glmer		glmmLasso	bGLMM (EM)	bGLMM (REML)
σ_b	p	mse_{σ_b}	mse_{σ_b}	n.c.	mse_{σ_b}	n.c.	mse_{σ_b}	mse_{σ_b}	mse_{σ_b}
0.4	3	0.004	0.004	0	0.004	0	0.007	0.040	0.003
0.4	5	0.004	0.004	0	0.004	0	0.007	0.040	0.003
0.4	10	0.004	0.004	0	0.004	0	0.007	0.040	0.003
0.4	20	0.004	0.005	0	0.005	0	0.006	0.040	0.003
0.4	50	0.005	0.007	0	0.007	0	0.007	0.040	0.004
0.8	3	0.010	0.010	0	0.010	0	0.010	0.141	0.010
0.8	5	0.010	0.010	0	0.010	0	0.010	0.141	0.010
0.8	10	0.010	0.010	0	0.010	0	0.010	0.141	0.010
0.8	20	0.010	0.010	0	0.010	0	0.010	0.141	0.010
0.8	50	0.010	0.011	0	0.011	0	0.010	0.141	0.010
1.6	3	0.067	0.029	7	0.031	1	0.033	1.268	0.040
1.6	5	0.047	0.029	7	0.031	1	0.033	1.268	0.040
1.6	10	0.034	0.029	7	0.031	1	0.033	1.268	0.040
1.6	20	0.033	0.029	7	0.031	3	0.033	1.268	0.040
1.6	50	0.033	0.029	7	0.032	3	0.033	1.269	0.040

Table 2: Mean squared errors for σ_b for the `glmmLasso` and alternative approaches on Poisson data

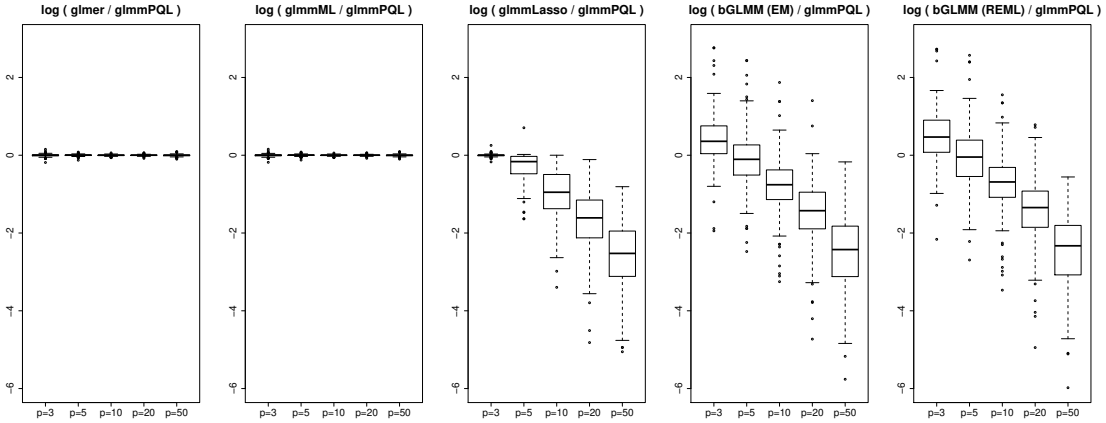

Figure 1: Boxplots of $\log(\text{mse}_{\boldsymbol{\beta}}(\cdot)/\text{mse}_{\boldsymbol{\beta}}(\texttt{glmmPQL}))$ for the `glmmLasso` and alternative approaches on Poisson data

Bernoulli Link The underlying model is the random intercept Bernoulli model

$$\eta_{it} = \sum_{j=1}^{p} x_{itj}\beta_j + b_i, \quad i = 1,\ldots,40, \quad t = 1,\ldots,10$$

$$E[y_{it}] = \frac{\exp(\eta_{it})}{1+\exp(\eta_{it})} := \pi_{it} \qquad y_{it} \sim \text{B}(1,\pi_{it})$$

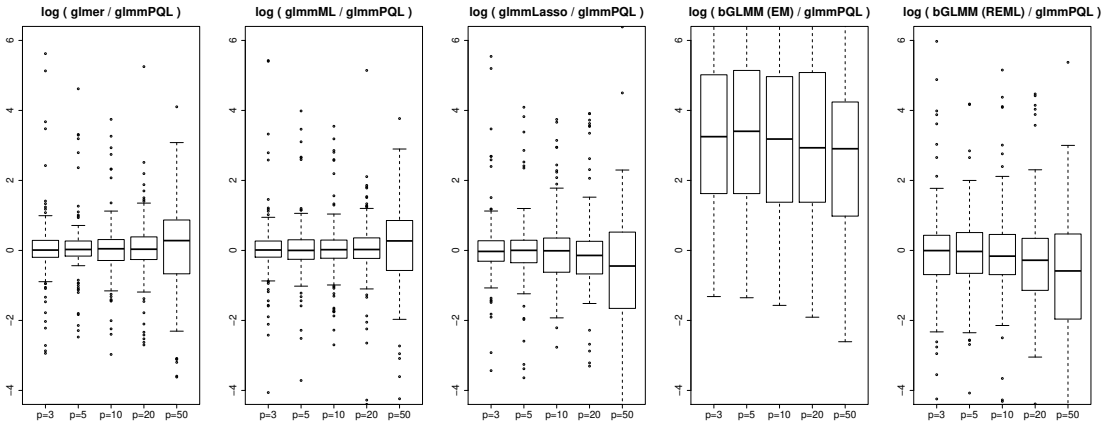

Figure 2: Boxplots of $\log(\text{mse}_{\sigma_b}(\cdot)/\text{mse}_{\sigma_b}(\texttt{glmmPQL}))$ for the `glmmLasso` and alternative approaches on Poisson data

with linear effects given by $\beta_1 = -5, \beta_2 = -10, \beta_3 = 15$ and $\beta_j = 0, j = 4, \ldots, 50$. Again we choose the different settings $p = 3, 5, 10, 20, 50$. For $j = 1, \ldots, 50$ the vectors $\mathbf{x}_{it}^T = (x_{it1}, \ldots, x_{it50})$ have been drawn independently with components following a uniform distribution within the interval $[-0.1, 0.1]$. The number of observations remains $n = 40, T_i := T = 10, \forall i = 1, \ldots, n$. The random effects and the noise variable have been specified by $b_i \sim N(0, \sigma_b^2)$ with $\sigma_b = 0.4, 0.8, 1.6$.

Again, we evaluate the performance of estimators separately for structural components and variance and compare the results of our `glmmLasso` algorithm with the alternative approaches mentioned for the Poisson case, based on the introduced goodness-of-fit criteria.

		`glmmPQL`	`glmmML`	`glmer`	`glmmLasso`			`bGLMM` (EM)			`bGLMM` (REML)		
σ_b	p	$\text{mse}_{\boldsymbol{\beta}}$	$\text{mse}_{\boldsymbol{\beta}}$	$\text{mse}_{\boldsymbol{\beta}}$	$\text{mse}_{\boldsymbol{\beta}}$	f.p.	f.n.	$\text{mse}_{\boldsymbol{\beta}}$	f.p.	f.n.	$\text{mse}_{\boldsymbol{\beta}}$	f.p.	f.n.
0.4	3	13.631	14.366	14.347	16.213	0	0.16	37.237	0	0.77	37.560	0	0.74
0.4	5	21.167	22.263	22.224	23.204	0.39	0.31	37.505	0.01	0.77	37.828	0.01	0.74
0.4	10	43.619	45.831	45.736	32.275	0.94	0.37	38.170	0.03	0.77	38.713	0.04	0.74
0.4	20	95.141	99.897	99.645	38.982	0.87	0.50	39.451	0.07	0.77	39.992	0.08	0.74
0.4	50	330.687	345.939	344.743	45.083	0.76	0.63	41.952	0.15	0.76	42.901	0.17	0.74
0.8	3	14.655	15.178	15.177	17.344	0	0.16	38.803	0	0.67	38.052	0	0.67
0.8	5	22.536	24.040	24.021	25.041	0.48	0.30	39.206	0.01	0.67	38.409	0.01	0.67
0.8	10	44.875	49.124	49.054	35.812	0.95	0.47	42.370	0.08	0.67	41.173	0.08	0.67
0.8	20	96.779	107.291	107.064	41.011	0.78	0.55	45.176	0.15	0.66	44.081	0.16	0.66
0.8	50	334.792	369.779	368.445	53.202	0.79	0.72	58.722	0.44	0.64	55.847	0.44	0.64
1.6	3	19.432	20.414	20.425	24.610	0	0.27	42.226	0	0.61	41.843	0	0.61
1.6	5	29.360	32.072	32074	29.565	0.54	0.27	42.805	0.01	0.61	42.363	0.01	0.61
1.6	10	56.144	63.519	63.515	42.283	1.26	0.39	44.694	0.05	0.61	44.159	0.05	0.61
1.6	20	125.207	143.594	143.415	48.668	0.81	0.50	49.666	0.15	0.61	48.524	0.14	0.61
1.6	50	488.798	542.524	538.381	60.148	0.93	0.60	58.913	0.35	0.60	56.880	0.33	0.60

Table 3: Mean squared errors for $\boldsymbol{\beta}$ for the `glmmLasso` and alternative approaches on Bernoulli data

The results for varying number p of covariates $x_{it1}, \ldots, x_{itp}$ and different random effects variances $\boldsymbol{\sigma}$ are summarized in Table 3 and 4. In general the results for the Bernoulli case have deteriorated for all different approaches, in particular in terms of $\text{mse}_{\boldsymbol{\beta}}$. But the general trend,

σ_b	p	glmmPQL mse_{σ_b}	glmmML mse_{σ_b}	glmer mse_{σ_b}	glmmLasso mse_{σ_b}	bGLMM (EM) mse_{σ_b}	bGLMM (REML) mse_{σ_b}
0.4	3	0.063	0.063	0.062	0.064	0.261	0.065
0.4	5	0.063	0.063	0.063	0.064	0.261	0.065
0.4	10	0.063	0.063	0.062	0.065	0.263	0.066
0.4	20	0.064	0.065	0.065	0.064	0.265	0.066
0.4	50	0.092	0.087	0.086	0.065	0.267	0.067
0.8	3	0.041	0.046	0.044	0.043	0.951	0.069
0.8	5	0.041	0.046	0.044	0.043	0.954	0.069
0.8	10	0.041	0.045	0.045	0.044	0.962	0.069
0.8	20	0.042	0.047	0.046	0.044	0.976	0.068
0.8	50	0.071	0.072	0.069	0.044	1.032	0.065
1.6	3	0.086	0.091	0.088	0.100	5.676	0.330
1.6	5	0.085	0.093	0.089	0.099	5.680	0.330
1.6	10	0.079	0.094	0.089	0.098	5.685	0.326
1.6	20	0.079	0.110	0.100	0.097	5.718	0.321
1.6	50	0.228	0.316	0.277	0.097	5.756	0.310

Table 4: Mean squared errors for σ_b for the `glmmLasso` and alternative approaches on Bernoulli data

that, in case of many covariates, the $\boldsymbol{\beta}$-fit that is achieved using the `glmmLasso` algorithm outperforms the fit obtained by the standard R functions, can still be observed.

Compared to Poisson case, the fit obtained by `glmmLasso` algorithm has even slightly improved with regard to both boosting approaches.

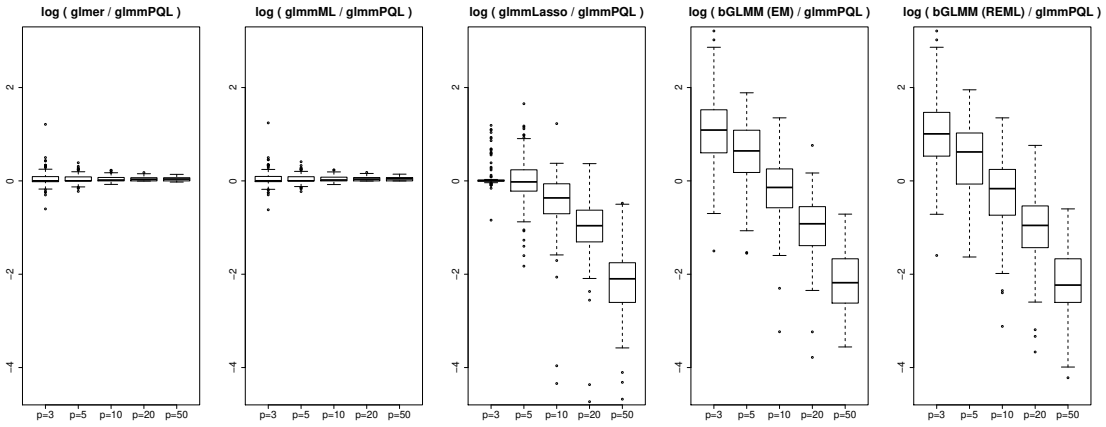

Figure 3: Boxplots of $\log(\text{mse}_{\boldsymbol{\beta}}(\cdot)/\text{mse}_{\boldsymbol{\beta}}(\texttt{glmmPQL}))$ for the `glmmLasso` and alternative approaches on Bernoulli data

4 Applications to Real Data

In the following sections we will apply our lasso method on different real data sets and compare the results with other approaches. The tuning parameters λ have been chosen via 5-fold cross validation. Standard errors for fixed effects and random effects variance components can be obtained by simulation-based parametric bootstrap evaluations, see Appendix B.

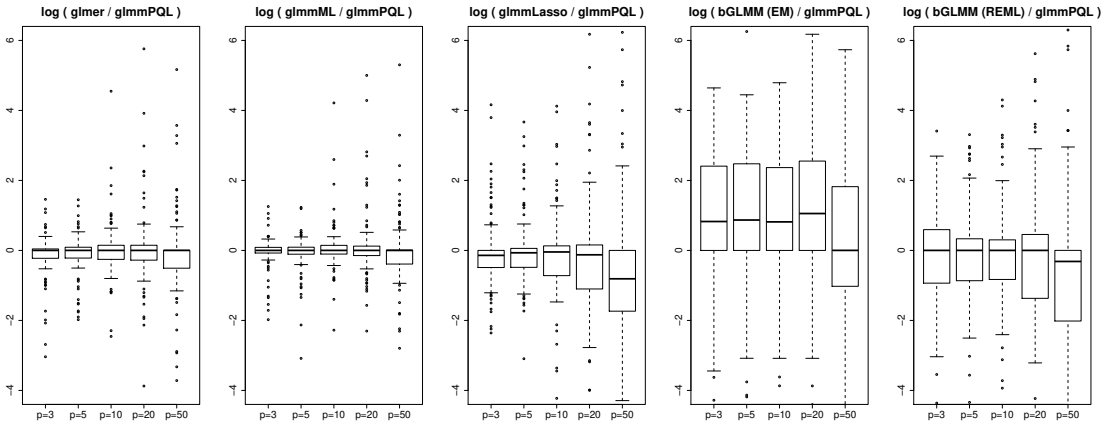

Figure 4: Boxplots of $\log(\text{mse}_{\sigma_b}(\cdot)/\text{mse}_{\sigma_b}(\texttt{glmmPQL}))$ for the `glmmLasso` and alternative approaches on Bernoulli data

4.1 The German Bundesliga

In the study the effect of team specific influence variables on the sportive success of the 18 soccer clubs of Germany's first soccer division, the Bundesliga, has been investigated for the last three seasons 2007/2008 to 2009/2010. The response variable is the number of points on which the league's form table is based. Each team gets three points for wins, one point for every draw and no points for defeats. A brief description of the team specific covariates in the data can be found in Table 9.

Covariate	Description
ball possession	average percentage of ball possession per game
tackle	average percentage of tackles won per game
unfairness	average number of unfairness points per game (1 point for yellow card, 3 points for second yellow card, 5 points for red card)
transfer spendings	money spent for new players during a season (in Euro)
transfer receipts	money earned through player transfers during a season (in Euro)
attendance	average attendance during a season
sold out	number of ticket sold outs during a season

Table 5: Description of covariates for the German Bundesliga data

Earlier studies have shown that the effect of the variable "transfer spendings" is parabolic (see Groll and Tutz, 2011). Therefore, we allowed "transfer spendings" to have a quadratic effect. Due to the very different ranges of values covariates have been standardized. The

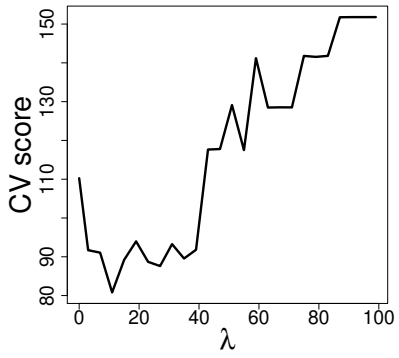

Figure 5: 5-fold cross-validation scores for the `glmmLasso` as function of penalty parameter λ for the German Bundesliga data

corresponding linear mixed model has the form

$$\begin{aligned} g(\mu_{it}) \quad = \quad & \beta_0 + \text{transfer spending}_{it}\beta_1 + \text{transfer spending}_{it}^2\beta_2 + \text{unfairness}_{it}\beta_3 \\ & + \text{transfer receipts}_{it}\beta_4 + \text{ball possession}_{it}\beta_5 + \text{tackles}_{it}\beta_6 \\ & + \text{attendance}_{it}\beta_7 + \text{sold out}_{it}\beta_8 + b_i, \end{aligned}$$

where μ_{it} denotes the expected number of points for soccer team i in season t and $b_i \sim N(0, \sigma_b^2)$ are team-specific random intercepts.

We fit an over-dispersed Poisson model with natural link and estimate the over-dispersion parameter Φ by use of Pearson residuals $\hat{r}_{it} = y_{it} - \hat{\mu}_{it}/(v(\hat{\mu}_{it}))^{\frac{1}{2}}$ by

$$\hat{\Phi} = \frac{1}{N - \text{df}} \sum_{i=1}^{n} \sum_{t=1}^{T_i} \hat{r}_{it}^2, \qquad N = \sum_{i=1}^{n} T_i, \tag{10}$$

where the degrees of freedom (df) correspond to the trace of the hat-matrix.

For selection of the penalty parameter λ for the `glmmLasso` 5-fold cross-validation was employed. The corresponding validation scores of prediction errors, based on the deviance, can be found in Figure 5. The cross-validation curve indicates that penalization clearly improves over ordinary fitting procedures that are obtained for $\lambda = 0$.

The results for the estimation of fixed effects, over-dispersion parameter $\hat{\Phi}$ and $\hat{\sigma}_b$ for the `glmmPQL` function and for the `glmmLasso` algorithm are given in Table 6 and the corresponding coefficient built-ups are illustrated in Figure 6. The `glmmLasso` algorithm suggests that "unfairness", "ball possession" and "tackles" are not needed in the predictor, which are all three far away from significance concerning the standard errors of the `glmmPQL` function given in brackets.

	glmmPQL	glmmLasso
intercept	3.860 (0.029)	3.858 (0.031)
transfer spendings	0.179 (0.061)	0.174 (0.083)
transfer spendings^2	-0.046 (0.015)	-0.043 (0.023)
unfairness	-0.022 (0.028)	-
transfer receipts	0.043 (0.033)	0.047 (0.030)
ball possession	0.008 (0.043)	-
tackles	0.015 (0.041)	-
attendance	0.059 (0.031)	0.068 (0.031)
sold out	0.113 (0.033)	0.120 (0.031)
$\hat{\sigma}_b$	0.000	0.005 (0.012)
$\hat{\Phi}$	1.637	1.394

Table 6: Estimates for the German Bundesliga data with `glmmPQL` function and `glmmLasso` algorithm (standard errors in brackets)

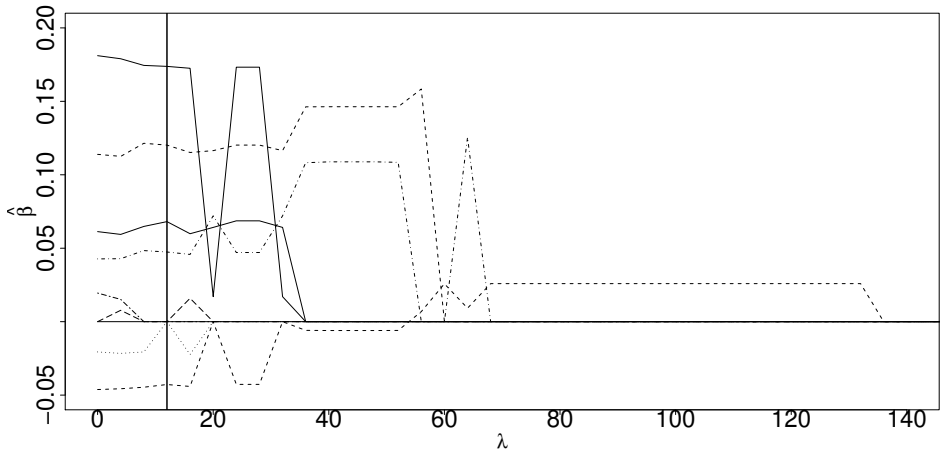

Figure 6: Coefficient built-ups for the `glmmLasso` for the German Bundesliga data; the optimal value of the penalty parameter λ is shown by the vertical line

With variable selection the estimated dispersion parameter is not far away from one, so that the Poisson model seems adequate. The `glmmPQL` function provides a very low standard deviation ($\hat{\sigma}_b$=0.000002) of the random intercepts, while the `glmmLasso` model leads to results that support the application of a random effects model, indicating that each soccer team has an individual bases level of points. In Figure 7 the quadratic effect of the variable "transfer spendings" is presented. Both approaches estimate very similar functions.

In addition we show the estimated random intercepts of the `glmmLasso` functions for the 23 different soccer teams, that played in the German Bundesliga during the seasons 2007/2008 to 2009/2010. They can be seen as representing the team-specific playing ability that is not covered by the explanatory variables (see Table 7).

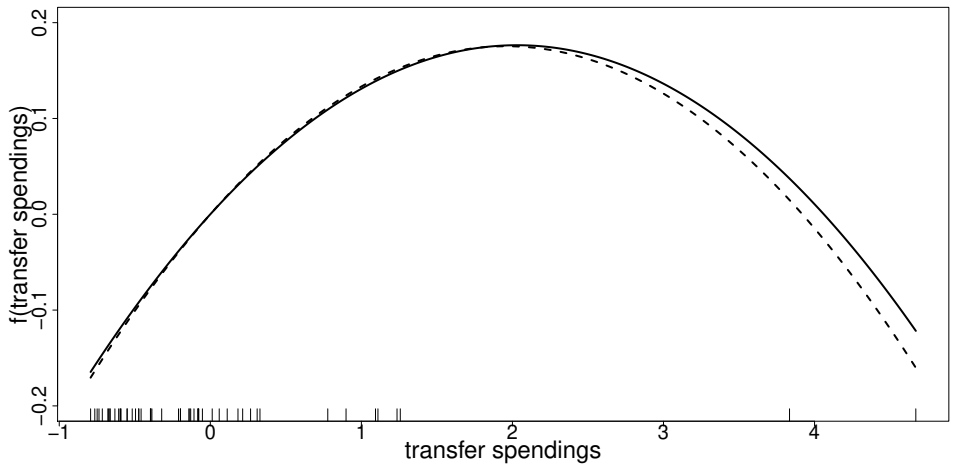

Figure 7: Estimated smooth effects computed with the `glmmPQL` model (dashed line) and the `glmmLasso` model (solid line) for the German Bundesliga data

For example the VfL Wolfsburg owns a small soccer stadium with a low number of ticket sold outs and was nevertheless rather successful in the last three years, so as a consequence its team-specific parameter is quite enhanced. The reverse effect could be observed e.g. for the FC Bayern München. The club has earned by far the most points on average, but as it exhibits a rather high average attendance, with the stadium being permanently sold out, it got a relatively low random intercept, though being the most successful club in the league during the last three seasons.

4.2 CD4 Aids Study

The data were collected within the Multicenter AIDS Cohort Study (MACS). In the study about 5000 infected gay or bisexual men from Baltimore, Pittsburgh, Chicago and Los Angeles have been observed since 1984 (see Kaslow et al., 1987; Zeger and Diggle, 1994). The human immune deficiency virus (HIV) causes AIDS by attacking an immune cell called the CD4+ cell which coordinates the body's immunoresponse to infectious viruses and hence reduces a person's resistance against infection. According to Diggle et al. (2002) an uninfected individual has around 110 cells per milliliter of blood and since the number of CD4+ cells decreases with time from infection, one can use an infected person's CD4+ cell number to check disease progression. Within the MACS, $n = 369$ seroconverters with a total of $\sum_{i=1}^{n} T_i = 2376$ measurements were included with the number of CD4+ cells being the interesting response variable. Covariates include the time since seroconversion ranging from 3 years before to 6 years after seroconversion, packs of cigarettes a day, recreational drug use (yes/no), number of sexual partners, age and a mental illness score (cesd). For observation t of individual i, the

Team	$\hat{b}_i$ (`glmmLasso`)	$\hat{b}_i$ (`glmmPQL`)
VfL Wolfsburg	0.060	$2.66{\cdot}10^{-4}$
VfB Stuttgart	0.056	$2.76{\cdot}10^{-4}$
Bayer 04 Leverkusen	0.054	$2.52{\cdot}10^{-4}$
Werder Bremen	0.049	$2.23{\cdot}10^{-4}$
FSV Mainz 05	0.022	$7.70{\cdot}10^{-5}$
Hertha BSC	0.021	$9.90{\cdot}10^{-5}$
Karlsruher SC	0.019	$7.41{\cdot}10^{-5}$
Borussia Dortmund	0.016	$1.08{\cdot}10^{-4}$
FC Bayern München	0.011	$5.72{\cdot}10^{-5}$
Hannover 96	0.008	$3.83{\cdot}10^{-5}$
Energie Cottbus	0.007	$1.79{\cdot}10^{-5}$
FC Schalke 04	-0.002	$-8.64{\cdot}10^{-6}$
Eintracht Frankfurt	-0.006	$-2.18{\cdot}10^{-5}$
Hansa Rostock	-0.007	$2.59{\cdot}10^{-5}$
VfL Bochum	-0.014	$-7.22{\cdot}10^{-5}$
SC Freiburg	-0.015	$-5.93{\cdot}10^{-5}$
Arminia Bielefeld	-0.018	$-7.83{\cdot}10^{-4}$
MSV Duisburg	-0.020	$7.31{\cdot}10^{-5}$
1899 Hoffenheim	-0.032	$-1.67{\cdot}10^{-4}$
Hamburger SV	-0.037	$-2.52{\cdot}10^{-4}$
1. FC Nürnberg	-0.051	$-2.09{\cdot}10^{-4}$
FC Köln	-0.059	$-2.54{\cdot}10^{-4}$
Borussia M'gladbach	-0.062	$2.69{\cdot}10^{-4}$

Table 7: Estimated random intercepts for German Bundesliga teams using `glmmLasso`.

model that is considered has the form

$$\begin{aligned} g(\mu_{it}) \quad = \quad & \beta_0 + \text{time}_{it}\beta_1 + \text{time}_{it}^2\beta_2 + \text{time}_{it}^3\beta_3 + \text{time}_{it}^4\beta_4 + \text{drugs}_{it}\beta_5 \\ & + \text{partners}_{it}\beta_6 + \text{cigarettes}_{it}\beta_7 + \text{cesd}_{it}\beta_8 + \text{age}_{it}\beta_9 + b_i, \end{aligned}$$

with $b_i \sim N(0, \sigma_b^2)$. Again we fit an over-dispersed Poisson model with natural link while the over-dispersion parameter Φ is estimated using (10). Our main objective is the typical time course of CD4+ decay and the variability across subjects. Earlier studies (e.g. Tutz and Reithinger, 2007, Groll and Tutz, 2011) have shown, that the time effect is nonlinear, so we additionally considered some higher powers of "time".

The chosen penalty parameter λ for the `glmmLasso` again was rather small, $\lambda_{\text{opt}} = 21000$, and consequently almost all of the variables are included. The results for the `glmmLasso` algorithm and for the `glmmPQL` function are given in Table 8 and the corresponding coefficient

built-ups are illustrated in Figure 8. Both approaches yield very similar estimates. The incorporated selection procedure suggests that drug use and age are not needed in the predictor.

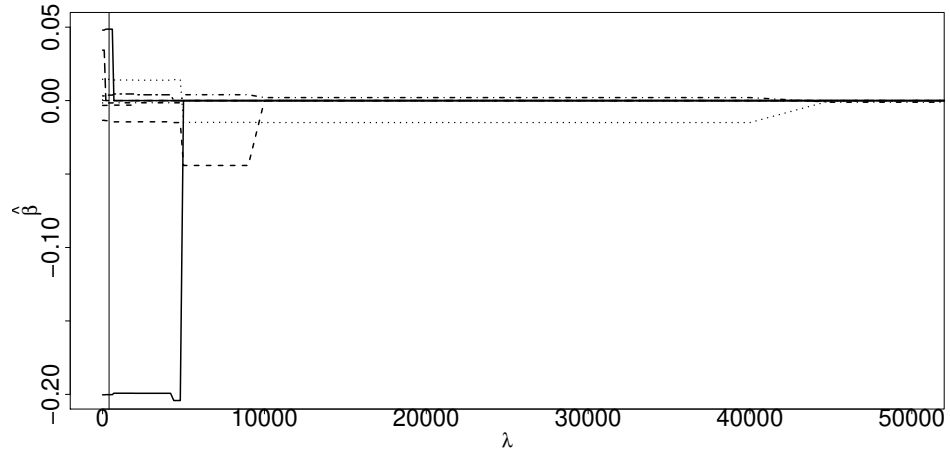

Figure 8: Coefficient built-ups for the `glmmLasso` for the CD4 data; the optimal value of the penalty parameter λ is shown by the vertical line

	`glmmPQL`	`glmmLasso`
Intercept	6.643 (0.028)	6.665 (0.026)
Time	-0.199 (0.011)	-0.200 (0.012)
Time^2	-0.014 (0.004)	-0.014 (0.004)
Time^3	0.014 (0.003)	0.014 (0.002)
Time^4	-0.002 (0.000)	-0.002 (0.000)
Drugs	0.029 (0.023)	-
Partners	0.004 (0.003)	0.004 (0.003)
Packs of Cigarettes	0.042 (0.009)	0.049 (0.007)
Mental illness score (cesd)	-0.003 (0.010)	-0.003 (0.001)
Age	0.000 (0.002)	-
$\hat{\sigma}_b$	0.298	0.252 (0.091)
$\hat{\Phi}$	63.439	76.943

Table 8: Estimates for the MACS with `glmmPQL` function and `glmmLasso` algorithm (standard deviations in brackets)

The smooth effect of time on CD4+ cell decay for our over-dispersed Poisson model together with the data is shown in Figure 9. Besides, we show the smooth effect obtained by a penalized basis function approach which is implemented in the `gamm` function of the R-package `mgcv` (Wood, 2006). All other variables have been kept constant at their means. Obviously the variable time has a negative effect on the CD4+ cell number.

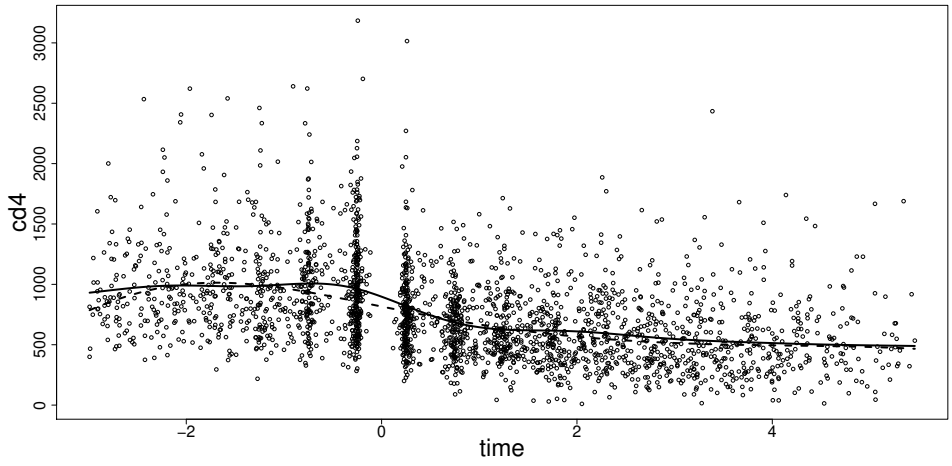

Figure 9: Smoothed time effect (CD4+ number of cells *versus* time) from MACS for `gamm` (solid line) and `glmmLasso` (dashed line).

4.3 Forest health Data

The forest health data has been considered in previous studies, for example in Kneib et al. (2009) and Tutz and Groll (2011). In this application, the health status of beeches at 83 observation plots located in a northern Bavarian forest district has been assessed in visual forest health inventories carried out between 1983 and 2004. Originally, the health status is classified on an ordinal scale, where the nine possible categories denote different degrees of defoliation. Figure 10 shows a histogram of the nine defoliation classes indicating that no trees were observed in the last two categories. We are now only interested in wether a tree is healthy or not, so we model the dichotomized response variable defoliation with categories 1 (not healthy; defoliation above or equal 12.5%) and 0 (healthy; no defoliation; 0.0%). In Kneib et al. (2009) a brief description of the covariates in the data set is presented, which is found in Table 9.

Covariate	Description
age	age of the tree in years (continuous, $7 \leq$ age ≤ 234)
elevation	elevation above sea level in meters (continuous, $250 \leq$ elevation ≤ 480)
inclination	inclination of slope in percent (continuous, $0 \leq$ inclination ≤ 46)
soil	depth of soil layer in centimeters (continuous, $9 \leq$ soil ≤ 51)
canopy	density of forest canopy in percent (continuous, $0 \leq$ canopy ≤ 1)
stand	type of stand (categorical, 1 = deciduous forest, -1 = mixed forest)
fertilisation	fertilisation (categorical, 1 = yes, -1 = no)
humus	thickness of humus layer in 5 categories (ordinal, higher categories represent higher proportions)
moisture	level of soil moisture (categorical, 1 = moderately dry, 2 = moderately moist, 3 = moist or temporary wet)
saturation	base saturation (ordinal, higher categories indicate higher base saturation)

Table 9: Description of covariates for the forest health data

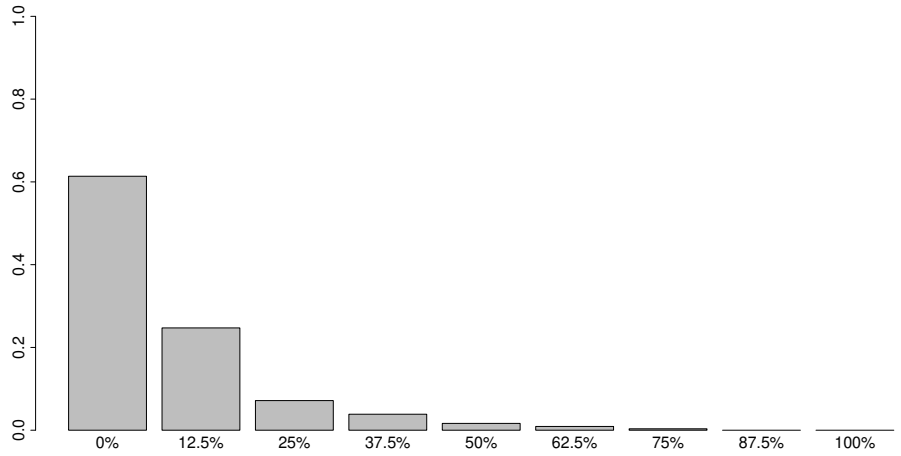

Figure 10: Relative frequencies of the nine defoliation classes for all observation plots and all time points for the forest health data

As Kneib et al. (2009) identified a nonlinear effect of "age", we again include some higher powers of "age" into our model, which results in the following predictor:

$$\begin{aligned} g(\pi_{it}) &= \beta_0 + \text{age}_{it}\beta_1 + \text{age}_{it}^2\beta_2 + \text{age}_{it}^3\beta_3 + \text{age}_{it}^4\beta_4 + \text{elevation}_{it}\beta_5 \\ &\quad + \text{inclination}_{it}\beta_6 + \text{soil}_{it}\beta_7 + \text{canopy}_{it}\beta_8 + \text{fertilisation}_{it}\beta_9 + \text{stand}_{it}\beta_{10} \\ &\quad + \text{humus0}_{it}\beta_{11} + \text{humus2}_{it}\beta_{12} + \text{humus3}_{it}\beta_{13} + \text{humus4}_{it}\beta_{14} + \text{saturation1}_{it}\beta_{15} \\ &\quad + \text{saturation3}_{it}\beta_{16} + \text{saturation4}_{it}\beta_{17} + \text{moisture1}_{it}\beta_{18} + \text{moisture3}_{it}\beta_{19} + b_i, \end{aligned}$$

where $\pi_{it} = \mu_{it}$ denotes the expected probability of defoliation for observation area i at time t and $b_i \sim N(0, \sigma_b^2)$ again represent cluster-specific random intercepts. We fit a binomial model with logit-link, building groups for the categorial variables "humus", "moisture" and "saturation". For this purpose we use the extended algorithm for categorical predictors from Section 3.3. The results for the parameter estimates can be found in Table 10 and the corresponding coefficient built-ups are illustrated in Figure 11.

The penalty parameter λ for the `glmmLasso` again was determined by 5-fold cross-validation on the interval $[0; 300]$. The chosen parameter was rather small, $\lambda_{\text{opt}} = 10$, indicating that penalization only slightly improves the fit compared to ordinary fitting procedures which are obtained for $\lambda = 0$ and consequently almost all of the variables are included. The smooth effect of age on tree defoliation for our binomial model with logit-link is shown in Figure 12, again compared to the smooth effect obtained by the penalized basis function approach using the `gamm` function. Obviously with increasing age of the trees the probability of defoliation increases in a non-linear fashion.

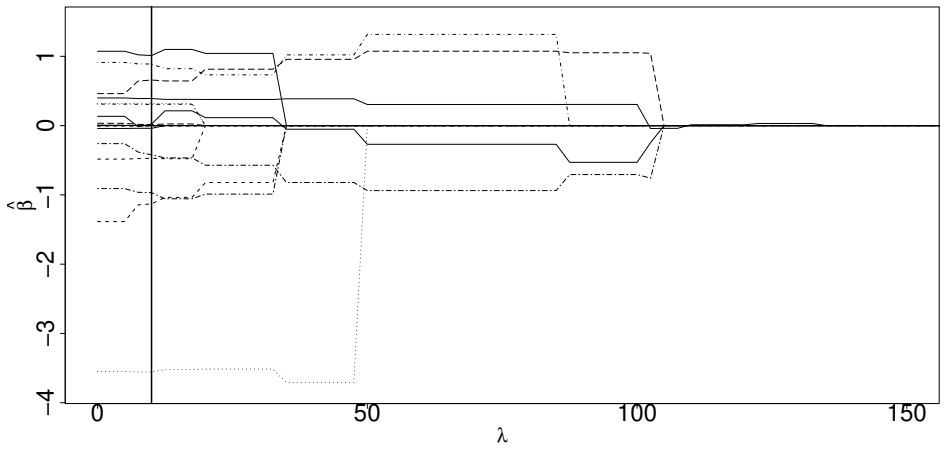

Figure 11: Coefficient built-ups for the `glmmLasso` for the forest health data; the optimal value of the penalty parameter λ is shown by the vertical line

	`glmmPQL`	`glmmLasso`
Intercept	-7.226 (2.719)	-5.353 (1.923)
age	0.401 (0.080)	0.391 (0.078)
age^2	-0.007 (0.001)	-0.006 (0.001)
age^3	0.000 (0.000)	0.000 (0.000)
age^4	0.000 (0.000)	0.000 (0.000)
elevation	0.006 (0.005)	-
inclination	0.005 (0.025)	-
soil	-0.043 (0.024)	-0.039 (0.026)
canopy	-3.550 (0.539)	-3.554 (0.568)
fertilisation	-1.422 (0.828)	-1.130 (0.773)
stand	0.934 (0.464)	0.889 (0.456)
humus0	-0.486 (0.155)	-0.472 (0.180)
humus2	0.316 (0.131)	0.323 (0.147)
humus3	0.313 (0.155)	0.306 (0.169)
humus4	0.036 (0.218)	0.011 (0.238)
saturation1	0.471 (0.533)	0.658 (0.483)
saturation3	-0.254 (0.557)	-0.422 (0.524)
saturation4	0.102 (0.699)	0.023 (0.652)
moisture1	-0.916 (0.522)	-0.968 (0.498)
moisture3	1.112 (0.379)	1.011 (0.365)
$\hat{\sigma}_b$	1.816	1.845 (0.177)

Table 10: Estimates for the forest health data

4.4 Jimma Infant Survival Study

The Jimma Infant Survival Differential Longitudinal Study is a cohort study investigating the live births which took place in the town of Jimma in Ethiopia during a one year period from September 1992 until September 1993. An extensive description can be found in Lesaffre et al. (1999). The study covers 8000 households with live births in the said period. Following Lesaffre

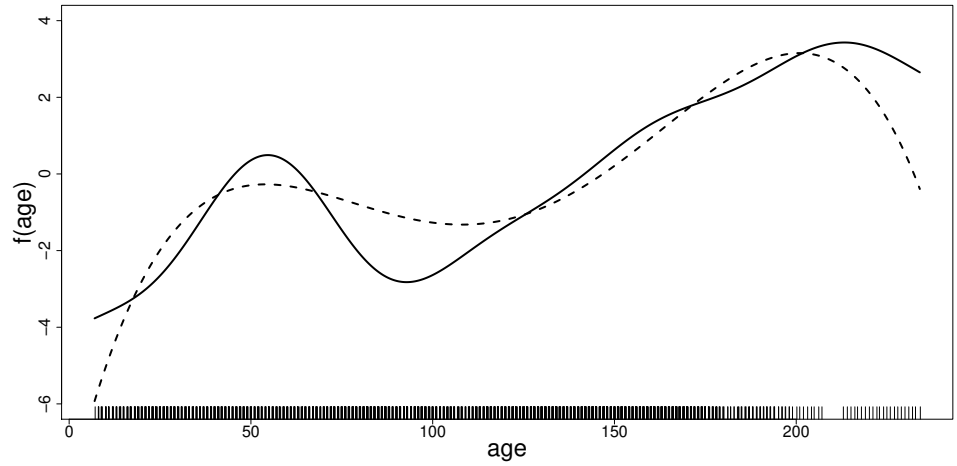

Figure 12: Smoothed age effect for the forest health data with `gamm` (solid line) and `glmmLasso` (dashed line).

et al. (1999) and Tutz and Reithinger (2007), 495 singleton live births have been considered and monitored for a one year period in order to determine the risk factors for infant mortality. A good indicator of a child's health status is the body weight. Hence, to determine possible influence factors on growth of the children, we use the (logarithmic) body weight (in kg) as response variable together with some socio-economic and demographic as well as some prenatal and delivery-related covariates. A brief description of all considered covariates can be found in Table 11.

Covariate	Description
age	age of the child in days (continuous, $0 \leq$ age ≤ 385)
ageM	age of the mother in years (continuous, $14 \leq$ ageM ≤ 50)
education	educational level of the mother (categorical, 1 = illiterate, 2 = read and write, 3 = elementary school, 4 = junior high school, 5 = high school, 6 = college and above)
delivery	place of delivery (categorical, 1 = hospital, 2 = health center, 3 = home)
visits	number of antenatal visits (categorical, 0, ≥ 1)
month	month of birth (categorical, 1 = Jan. - June, 0 = July - Dec.)
sex	sex of the child (categorical, 1 = male, 0 = female)
marital	marital status of mother (categorical, 1 = married, 2 = divorced, 3 = widowed, 4 = never married)
status	occupational status of mother (categorical, 1 = unemployed, 0 = employed)

Table 11: Description of covariates for the Jimma data

Tutz and Reithinger (2007) identified a nonlinear effect of "age", therefore we include also "age^2" into our model, resulting in the following predictor:

$$\begin{aligned} g(\mu_{it}) &= \beta_0 + \text{age}_{it}\beta_1 + \text{age}_{it}^2\beta_2 + \text{ageM}_{it}\beta_3 + \text{education1}_{it}\beta_4 + \text{education2}_{it}\beta_5 \\ &\quad + \text{education3}_{it}\beta_6 + \text{education4}_{it}\beta_7 + \text{education5}_{it}\beta_8 + \text{delivery1}_{it}\beta_9 \\ &\quad + \text{delivery2}_{it}\beta_{10} + \text{visits}_{it}\beta_{11} + \text{month}_{it}\beta_{12} + \text{sex}_{it}\beta_{13} + \text{marital1}_{it}\beta_{14} \\ &\quad + \text{marital2}_{it}\beta_{15} + \text{marital3}_{it}\beta_{16} + \text{status}_{it}\beta_{17} + b_{0i} + \text{age}_{it}b_{1i} + \text{age}_{it}^2 b_{2i}, \end{aligned}$$

where μ_{it} denotes the expected body weight of child i at time t and $\mathbf{b}_i = (b_{0i}, b_{1i}, b_{2i})^T \sim N(\mathbf{0}, \mathbf{Q})$ represent child-specific random intercepts and random slopes on age and squared age. The continuous variables age, squared age and age of the mother have been standardized. We fit a normal distribution model with log-link, building groups for the categorial variables "education", "delivery" and "marital". So again the extended algorithm for categorical predictors from Section 3.3 is required. The estimates for the standard deviations of the random effects for the standardized model are presented in Table 12.

	`glmmPQL`	`glmmLasso`
$\hat{\sigma}_{b_0}$	0.121	0.153 (0.046)
$\hat{\sigma}_{b_1}$	0.037	0.000 (0.051)
$\hat{\sigma}_{b_2}$	0.000	0.069 (0.045)

Table 12: Estimates for the standard deviations of the random effects for the Jimma data with `glmmPQL` function and `glmmLasso` algorithm (standard deviations in brackets)

The results for the estimated linear effects corresponding to the original scaling of the variables can be found in Table 13 and the corresponding coefficient built-ups are illustrated in Figure 14. The cross-validation score is plotted against the penalty parameter λ in Figure 13. Again penalization improves ordinary fitting procedures obtained for $\lambda = 0$ and a rather sparse model is chosen with a clearly non-linear influence of the child's age and a linear influence of the variables "delivery", "visits" and "sex".

Keeping all other variables constant at their means, the child-specific smooth effects of the children's age on the body weight is shown in Figure 15 and compared to the child-specific smooth effects obtained by the unregularized approach using the `glmmPQL` function, see Figure 16. It seems that there is somewhat more variation between the `glmmLasso` curves which may be due to the bigger variance estimate of the random intercept. As was to be expected, with increasing age of the children their body weight increases, at first relatively fast, but slowing down after the first 150 days. The main feature of the penalized approach is that variables that also turn out to be non-influential are automatically selected.

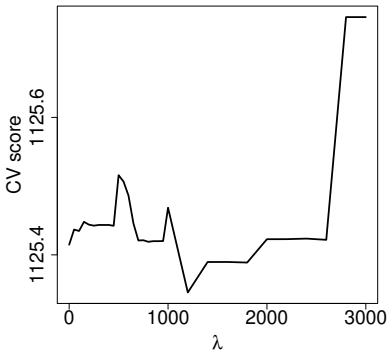

Figure 13: 5-fold cross-validation scores for the `glmmLasso` as function of penalty parameter λ for the Jimma data

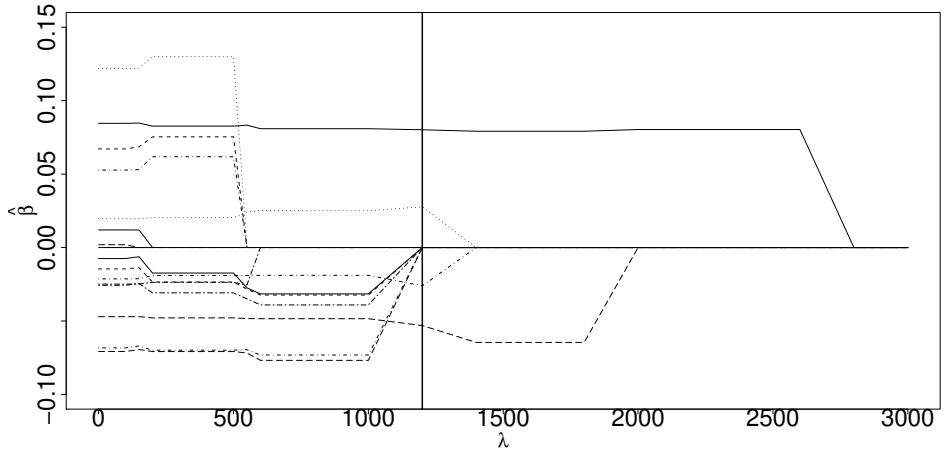

Figure 14: Coefficient built-ups for the `glmmLasso` for the Jimma data; the optimal value of the penalty parameter λ is shown by the vertical line

5 Concluding Remarks

Several procedures for variable selection based on L_1-penalties have been proposed. The procedures yield stable estimates in cases where methods that do not include variable selection typically fail because of the complexity of the fitting task. The method allows to include categorical predictors that are selected predictor or omitted as a whole predictor in the spirit of the group lasso. It is straightforward to extend the approach to include more complex penalty terms, for example, the elastic net penalty or hierarchical penalty terms as proposed by Zhao et al. (2009).

	glmmPQL	glmmLasso
Intercept	1.213 (0.052)	1.257 (0.045)
age	0.005 (0.000)	0.005 (0.000)
age^2	-0.000 (0.000)	-0.000 (0.000)
ageM	0.002 (0.001)	-
education1	-0.077 (0.041)	-
education2	-0.078 (0.042)	-
education3	-0.032 (0.041)	-
education4	-0.017 (0.041)	-
education5	-0.018 (0.040)	-
delivery1	0.021 (0.019)	0.028 (0.019)
delivery2	-0.024 (0.017)	-0.026 (0.016)
visits	-0.045 (0.013)	-0.053 (0.010)
month	-0.024 (0.012)	-
sex	0.081 (0.012)	0.080 (0.013)
marital1	0.057 (0.025)	-
marital2	0.104 (0.057)	-
marital3	0.056 (0.038)	-
occupational	0.001 (0.016)	-

Table 13: Estimated linear effects for the Jimma data with `glmmPQL` function and `glmmLasso` algorithm (standard deviations in brackets)

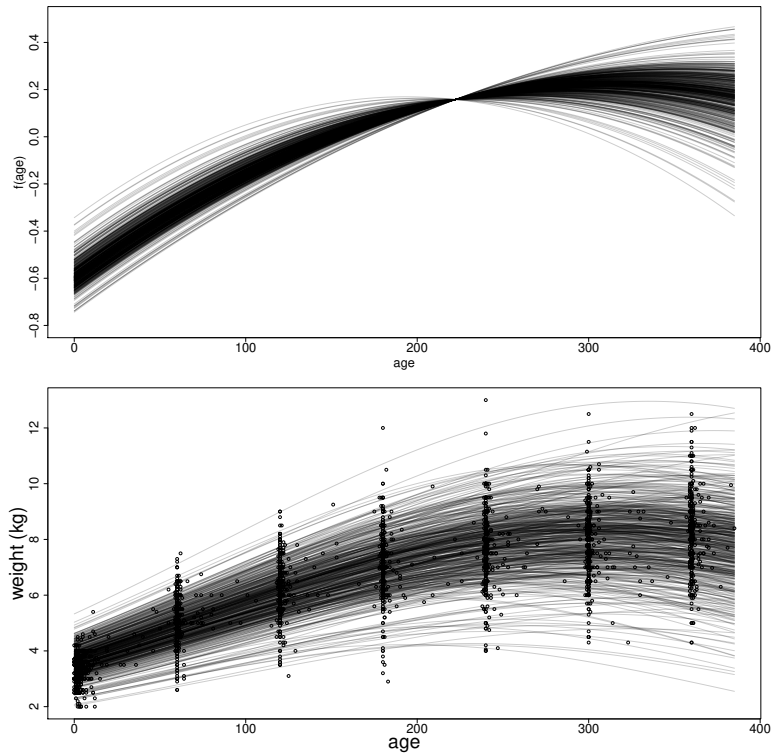

Figure 15: Individual smoothed age effects for the Jimma data on the predictor level (upper) and *versus* body weight (lower) for `glmmLasso` with slopes up to second potence of age.

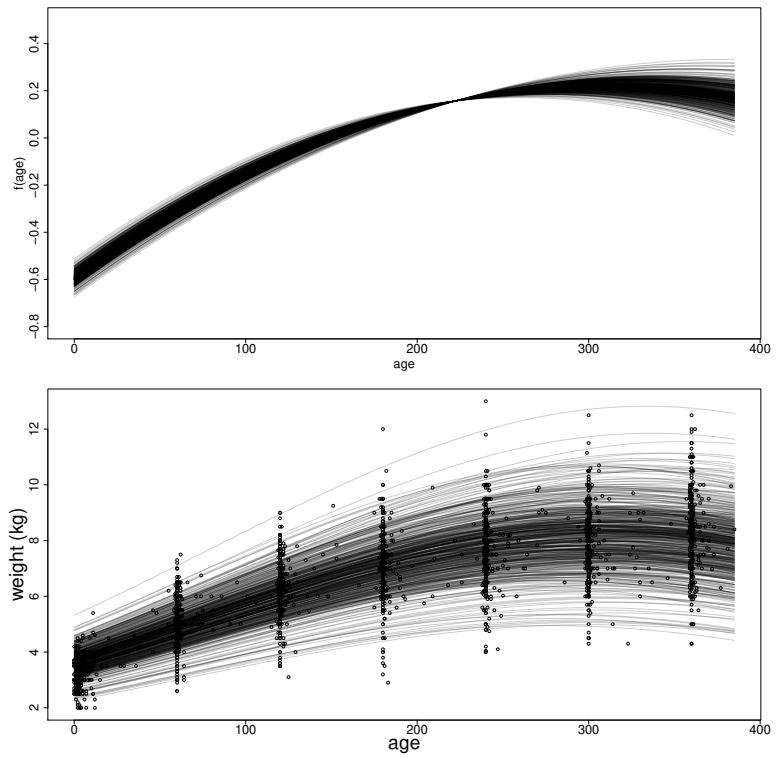

Figure 16: Individual smoothed age effects for the Jimma data on the predictor level (upper) and *versus* body weight (lower) for `glmmPQL` with slopes up to second potence of age.

Appendix

A Partition of Fisher Matrix

According to Fahrmeir and Tutz (2001) the penalized pseudo-Fisher matrix $\mathbf{F}^{\text{pen}}(\boldsymbol{\delta}) = \mathbf{A}^T\mathbf{W}(\boldsymbol{\delta})\mathbf{A} + \mathbf{K}$ can be partitioned into

$$\mathbf{F}^{\text{pen}}(\boldsymbol{\delta}) = \begin{bmatrix} \mathbf{F}_{\boldsymbol{\beta}\boldsymbol{\beta}} & \mathbf{F}_{\boldsymbol{\beta}1} & \mathbf{F}_{\boldsymbol{\beta}2} & \dots & \mathbf{F}_{\boldsymbol{\beta}n} \\ \mathbf{F}_{1\boldsymbol{\beta}} & \mathbf{F}_{11} & & & 0 \\ \mathbf{F}_{2\boldsymbol{\beta}} & & \mathbf{F}_{22} & & \\ \vdots & & & \ddots & \\ \mathbf{F}_{n\boldsymbol{\beta}} & 0 & & & \mathbf{F}_{nn} \end{bmatrix},$$

with single components

$$\begin{aligned} \mathbf{F}_{\boldsymbol{\beta}\boldsymbol{\beta}} &= -E\left(\frac{\partial^2 l^{\text{pen}}(\boldsymbol{\delta})}{\partial\boldsymbol{\beta}\partial\boldsymbol{\beta}^T}\right) = \mathbf{X}^T\mathbf{D}(\boldsymbol{\delta})\boldsymbol{\Sigma}(\boldsymbol{\delta})^{-1}\mathbf{D}(\boldsymbol{\delta})^T\mathbf{X}, \\ \mathbf{F}_{\boldsymbol{\beta}i} &= \mathbf{F}_{i\boldsymbol{\beta}}^T = -E\left(\frac{\partial^2 l^{\text{pen}}(\boldsymbol{\delta})}{\partial\boldsymbol{\beta}\partial\mathbf{b}_i^T}\right) = \mathbf{X}_i^T\mathbf{D}_i(\boldsymbol{\delta})\boldsymbol{\Sigma}_i(\boldsymbol{\delta})^{-1}\mathbf{D}_i(\boldsymbol{\delta})^T\mathbf{Z}_i, \\ \mathbf{F}_{ii} &= -E\left(\frac{\partial^2 l^{\text{pen}}(\boldsymbol{\delta})}{\partial\mathbf{b}_i\partial\mathbf{b}_i^T}\right) = \mathbf{Z}_i^T\mathbf{D}_i(\boldsymbol{\delta})\boldsymbol{\Sigma}_i(\boldsymbol{\delta})^{-1}\mathbf{D}_i(\boldsymbol{\delta})^T\mathbf{Z}_i + \mathbf{Q}^{-1}, \end{aligned}$$

and $\mathbf{D}_i(\boldsymbol{\delta}) = \partial h(\boldsymbol{\eta}_i)/\partial\boldsymbol{\eta}$, $\boldsymbol{\Sigma}_i(\boldsymbol{\delta}) = cov(\mathbf{y}_i|\beta, \mathbf{b}_i)$.

B Two Bootstrap approaches for GLMMs

The general idea of bootstrapping has been developed by Efron (1983, 1986). An extensive overview of the bootstrap and related methods for asserting statistical accuracy can be found in Efron and Tibshirani (1993). For GLMMs two main approaches are found in the literature. The first approach is to resample nonparametrically, which has been proposed e.g. by McCullagh (2000) and Davison and Hinkley (1997). They randomly sample groups of observations with replacement at the first stage and suggest various ways how to sample within the groups at the second stage. They showed that sometimes it can be useful to randomly resample groups at the first stage only and leave groups themselves unchanged, for example if there is a longitudinal structure in the data, see e.g. Shang and Cavanaugh (2008).

The second approach, on which the standard errors in Section 4 are based on, is to simulate parametric bootstrap samples following the parametric distribution family of the underlying model (compare Efron, 1982). Booth (1996) has extended the parametric approach from Efron (1982) to GLMMs to estimate standard errors for the fitted linear predictor $\hat{\boldsymbol{\eta}} = \mathbf{X}\hat{\boldsymbol{\beta}} + \mathbf{Z}\hat{\mathbf{b}}$ from

Section 2.

Analogously we can derive standard errors for the fixed effects estimate $\hat{\boldsymbol{\beta}}$ and for the estimated random effects variance components $\hat{\mathbf{Q}}$, respectively. Let $\{F_{\boldsymbol{\xi}} : \boldsymbol{\xi} \in \Xi\}$ denote the parametric distribution family of the underlying model, where $\boldsymbol{\xi}^T = (\boldsymbol{\beta}^T, \mathbf{b}^T, vec(\mathbf{Q})^T)$ is unknown. Here $vec(\mathbf{Q})$ denotes the column-wise vectorization of matrix $\mathbf{Q}$ to a column vector. Let $\hat{\boldsymbol{\xi}} = (\hat{\boldsymbol{\beta}}^T, \hat{\mathbf{b}}^T, vec(\hat{\mathbf{Q}})^T)$ denote the Lasso estimate of $\boldsymbol{\xi}$ for an already chosen penalty parameter λ on a certain data set. Now we can simulate new bootstrap data sets $(\mathbf{y}^*, \mathbf{b}^*)$ with respect to the distribution $F_{\hat{\boldsymbol{\xi}}}$, i.e. $(\mathbf{y}^*, \mathbf{b}^*) \sim F_{\hat{\boldsymbol{\xi}}}$. We repeat this procedure sufficiently often, say $B = 10.000$, and fit every new bootstrap data set $(\mathbf{y}^*_{(r)}, \mathbf{X}, \mathbf{W})$, $r = 1, \ldots, B$, with our `glmmLasso` algorithm. The new fits $\hat{\boldsymbol{\xi}}^*_{(r)}$ corresponding to the r-th new data set serve as bootstrap estimates and can be used to derive standard errors.

References

Bates, D. and M. Maechler (2010). *lme4: Linear mixed-effects models using S4 classes*. R package version 0.999375-34.

Bondell, H. D., A. Krishna, and S. Ghosh (2010). Joint variable selection of fixed and random effects in linear mixed-effects models. *Biometrics 66*, 1069–1077.

Booth, J. G. (1996). Bootstrap methods for generalized mixed models with applications to small area estimation. In G. U. H. Seeber, B. J. Francis, R. Hatzinger, and G. Steckel-Berger (Eds.), *Statistical Modelling*, Volume 104, pp. 43–51. New York: Springer.

Booth, J. G. and J. P. Hobert (1999). Maximizing generalized linear mixed model likelihoods with an automated Monte Carlo EM algorithm. *J. R. Statist. Soc B 61*, 265–285.

Breiman, L. (1996). Heuristics of instability and stabilization in model selection. *Annals of Statistics 6*, 2350–2383.

Breiman, L. (1998). Arcing classifiers. *Annals of Statistics 26*, 801–849.

Breslow, N. E. and D. G. Clayton (1993). Approximate inference in generalized linear mixed model. *Journal of the American Statistical Association 88*, 9–25.

Breslow, N. E. and X. Lin (1995). Bias correction in generalized linear mixed models with a single component of dispersion. *Biometrika 82*, 81–91.

Broström, G. (2009). *glmmML: Generalized linear models with clustering*. R package version 0.81-6.

Bühlmann, P. and T. Hothorn (2007). Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science 22*, 477–522.

Bühlmann, P. and B. Yu (2003). Boosting with the L2 loss: Regression and classification. *Journal of the American Statistical Association 98*, 324–339.

Candes, E. and T. Tao (2007). The dantzig selector: Statistical estimation when p is much larger than n. *The Annals of Statistics 35*, 2313–2351.

Davison, A. C. and D. V. Hinkley (1997). *Bootstrap Methods and Their Application*. Cambridge: Cambridge University Press.

Diggle, P. J., P. Heagerly, K. Y. Liang, and S. L. Zeger (2002). *Analysis of Longitudinal Data*. Oxford: Oxford University Press.

Efron, B. (1982). *The jackknife, the bootstrap and other resampling plans*, Volume 38. SIAM: CBMS-NSF Regional Conference Series in Applied Mathematics.

Efron, B. (1983). Estimating the error rate of a prediction rule: improvement on crossvalidation. *Journal of the American Statistical Association 78*, 316–331.

Efron, B. (1986). How biased is the apparent error rate of a prediction rule? *Journal of the American Statistical Association 81*, 461–470.

Efron, B., T. Hastie, I. Johnstone, and R. Tibshirani (2004). Least angle regression. *Annals of Statistics 32*, 407–499.

Efron, B. and R. Tibshirani (1993). *An Introduction to the Bootstrap*. New York: Chapman & Hall.

Fahrmeir, L. and S. Lang (1999). Bayesian inference for generalized additive mixed models based on Markov random field priors. *Applied Statistics* (to appear).

Fahrmeir, L. and G. Tutz (2001). *Multivariate Statistical Modelling Based on Generalized Linear Models* (2nd ed.). New York: Springer-Verlag.

Fan, J. and R. Li (2001). Variable selection via nonconcave penalize likelihood and its oracle properties. *Journal of the American Statistical Association 96*, 1348–1360.

Freund, Y. and R. E. Schapire (1996). Experiments with a new boosting algorithm. In *Proceedings of the Thirteenth International Conference on Machine Learning*, pp. 148–156. San Francisco, CA: Morgan Kaufmann.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of Statistics 29*, 337–407.

Genkin, A., D. Lewis, and D. Madigan (2007). Large-scale bayesian logistic regression for text categorization. *Technometrics 49*, 291–304.

Goeman, J. J. (2010). L_1 Penalized Estimation in the Cox Proportional Hazards Model. *Biometrical Journal 52*, 70–84.

Groll, A. and G. Tutz (2011). Variable Selection for Generalized Additive Mixed Models by Likelihood-based Boosting. Technical Report, Ludwig-Maximilians-University.

Gui, J. and H. Z. Li (2005). Penalized cox regression analysis in the high-dimensional and low-sample size settings, with applications to microarray gene expression data. *21*, 3001–3008.

Hastie, T., S. Rosset, R. Tibshirani, and J. Zhu (2004). The entire regularization path for the support vector machine. *Journal of Machine Learning Research 5*, 1391–1415.

James, G. M. and P. Radchenko (2009). A generalized dantzig selector with shrinkage tuning. *Biometrika 96,2*, 323–337.

Kaslow, R. A., D. G. Ostrow, R. Detels, J. P. Phair, B. Polk, and C. R. Rinaldo (1987). The multicenter aids cohort study: rationale, organization and selected characteristic of the participants. *American Journal of Epidemiology 126*, 310–318.

Kim, Y. and J. Kim (2004). Gradient lasso for feature selection. In *Proceedings of the 21st International Conference on Machine Learning*, pp. 473–480. Volume 69 of ACM International Conference Proceeding Series.

Kneib, T., T. Hothorn, and G. Tutz (2009). Variable selection and model choice in geoadditive regression. *Biometrics 65*, 626–634.

Lesaffre, E., M. Asefa, and G. Verbeke (1999). Assessing the godness-of-fit of the laird and ware model - an example: The jimma infant survival differential longitudinal study. *Statistics in Medicine 18*, 835–854.

Lin, X. and N. E. Breslow (1996). Bias correction in generalized linear mixed models with multiple components of dispersion. *Journal of the American Statistical Association 91*, 1007–1016.

Littell, R., G. Milliken, W. Stroup, and R. Wolfinger (1996). *SAS System for Mixed Models*. Cary, NC: SAS Institue Inc.

McCullagh, P. (2000). Re-sampling and exchangeable arrays. *Bernoulli 6*, 303–322.

McCulloch, C. E. and S. R. Searle (2001). *Generalized, Linear and Mixed Models*. New York: Wiley.

Meier, L., S. Van de Geer, and P. Bhlmann (2008). The group lasso for logistic regression. *Journal of the Royal Statistical Society B 70*, 53–71.

Ni, X., D. Zhang, and H. H. Zhang (2010). Variable Selection for Semiparametric Mixed Models in Longitudinal Studies. *Biometrics 66*, 79–88.

Osborne, M., B. Presnell, and B. Turlach (2000). On the lasso and its dual. *Journal of Computational and Graphical Statistics*.

Park, M. Y. and T. Hastie (2007). L1-regularization path algorithm for generalized linear models. *Journal of the Royal Statistical Society Series B 19*, 659–677.

Pinheiro, J. C. and D. M. Bates (2000). *Mixed-Effects Models in S and S-Plus*. New York: Springer.

Radchenko, P. and G. M. James (2008). Variable Iinclusion and Shrinkage Algorithms. *Journal of the American Statistical Association 103*, 1304–1315.

Schall, R. (1991). Estimation in generalised linear models with random effects. *Biometrika 78*, 719–727.

Segal, M. R. (2006). Microarray gene expression data with linked survival phenotypes: Diffuse large-b-cell lymphoma revisited. *Biostatistics 7*, 268–285.

Shang, J. and J. E. Cavanaugh (2008). Bootstrap variants of the akaike information criterion for mixed model selection. *Computational Statistics & Data Analysis 52*, 2004–2021.

Shevade, S. K. and S. S. Keerthi (2003). A simple and efficient algorithm for gene selection using sparse logistic regression. *Bioinformatics 19*, 2246–2253.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society B 58*, 267–288.

Tibshirani, R. (1997). The lasso method for variable selection in the cox model.

Tutz, G. and A. Groll (2010). Generalized Linear Mixed Models Based on Boosting. In T. Kneib and G. Tutz (Eds.), *Statistical Modelling and Regression Structures - Festschrift in the Honour of Ludwig Fahrmeir*. Physica.

Tutz, G. and A. Groll (2011). Binary and Ordinal Random Effects Models Including Variable Selection. Technical Report, LMU Munich.

Tutz, G. and F. Reithinger (2007). Flexible semiparametric mixed models. *Statistics in Medicine 26*, 2872–2900.

Venables, W. N. and B. D. Ripley (2002). *Modern Applied Statistics with S* (Fourth ed.). New York: Springer.

Vonesh, E. F. (1996). A note on the use of laplace's approximatio for nonlinear mixed-effects models. *Biometrika 83*, 447–452.

Wolfinger, R. and M. O'Connell (1993). Generalized linear mixed models; a pseudolikelihood approach. *Journal Statist. Comput. Simulation 48*, 233–243.

Wolfinger, R. W. (1994). Laplace's approximation for nonlinear mixed models. *Biometrika 80*, 791–795.

Wood, S. N. (2006). *Generalized Additive Models: An Introduction with R*. London: Chapman & Hall.

Yuan, M. and Y. Lin (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society B 68*, 49–67.

Zeger, S. L. and P. J. Diggle (1994). Semi-parametric models for longitudinal data with application to CD4 cell numbers in HIV seroconverters. *Biometrics 50*, 689–699.

Zhao, P., G. Rocha, and B. Yu (2009). The composite absolute penalties family for grouped and hierarchical variable selection. *Annals of Statistics 37*, 3468–3497.

Zou, H. and T. Hastie (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society B 67*, 301–320.

Zou, H. and T. Hastie (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association 101*, 1418–1429.