variable screening and parameter estimation for high-dimensional generalized linear mixed models using $\ell_1$-penalization

Jürg Schelldorfer and Peter Bühlmann

1. Overview

<table>
<thead>
<tr>
<th>$n &gt; p$</th>
<th>$n \ll p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generalized Linear Models (GLMs)</td>
<td>MLE [glm]</td>
</tr>
<tr>
<td>Generalized Linear Mixed Models (GLMMs)</td>
<td>MLE [glmm]</td>
</tr>
</tbody>
</table>

$n$: number of observations
$p$: number of variables

2. Generalized Linear Mixed Models and $\ell_1$-Penalized Estimation

Classical Model Set-up

Notation:

$g = 1, \ldots, N$ independent groups/clusters/subjects

$j = 1, \ldots, n_g$ observations for group/clustersubject $g$

$n = \sum n_g$, total number of observations

$y$: $1 \times d$-dim response variable

$b$: $d \times (d + 1)$ (correlated) random effects

$\theta$: $d \times 1$ fixed effects parameter

$\phi$: dispersion parameter

$X$: $n \times p$ model matrix for $\beta$

$Z$: $n \times q$ model matrix for $b$

$\Sigma_b$: $q \times q$ covariance matrix, determined by $\theta$

Model Assumptions (MA 1):

- $y \mid b$ are independent for $i = 1, \ldots, n$
- $y \mid b$ has a density of the form $\exp \left\{ -\sum_{j=1}^{p} (x_j - b_j \xi_j) + c(y, \theta) \right\} \mu_i$, with $\mu_i = E[y \mid b]$
- $g \mu_i = \eta_i$ with $\eta = X \beta + Z b$
- $b \sim N_0(0, \Sigma_b)$ for $\theta \in \mathbb{R}^d$

Spherical Coordinates (Bates, 2011):

Write $\Sigma_b = \lambda \Sigma_0$ and define $u = \lambda u_0$ where $u \sim N_0(0, I_n)$.

Parameter Estimation (Bates, 2011):

$\hat{b}(\beta, \phi)_{MLE} = \arg \min_{\beta, \phi} -\log L(\beta, \phi, \theta)$

where $L(\beta, \phi, \theta)$ is the likelihood function.

High-dimensional Model Set-up

Model Assumptions (MA 2):

- $n = \sum n_g < p$
- $P$ true $\beta$ is sparse
- $d$ small, say $d \leq 10$

Goal: Assuming (MA 1) and (MA 2), estimate $\beta, \phi$ and predict $b$.

The GLMMLasso Estimator

To cope with high-dimensionality and to enforce sparsity, we use a Lasso-type penalty (Tibshirani, 1996).

Hence

$Q_{\lambda}(\beta, \phi) = \hat{b}(\beta, \phi)_{MLE} = \arg \min_{\beta, \phi} -\log L(\beta, \phi, \theta)$

and we can use the Laplace approximation to approximate the integral of $L(\beta, \phi)$ by a quadratic function, i.e.

$I = \int_{-\infty}^{\infty} e^{-\frac{1}{2}u^T\Sigma_b u} du = \int_{-\infty}^{\infty} e^{-\frac{1}{2}u^T\Sigma_b u} du$

where $u = \arg \min_u S(u)$ is the mode of $-S(u)$.

The Laplace approximation of $Q_{\lambda}(\beta, \phi)$ is

$Q_{\lambda}(\beta, \phi) = \hat{b}(\beta, \phi)_{MLE} + \frac{1}{2} u_0^T\Sigma_u^{-1} u_0$

where

$S_{\lambda}(\beta, \phi) = \log L(\beta, \phi, \theta)$

and $v(\cdot)$ is the variance function.

The GLMMLasso estimator is defined by

$\hat{b}(\beta, \phi)_{GLMMLasso} = \arg \min_{\beta, \phi} -\log L(\beta, \phi, \theta)$

This is a high-dimensional, non-convex optimization problem!

3. Computational Algorithm

To calculate the GLMMLasso estimator (2), we use coordinatewise optimization with inexact line search, i.e. optimizing $Q_{\lambda}(\beta, \phi)$ with respect to one coordinate keeping all other coordinates fixed (Tseng and Yun, 2009). An overview of the algorithm may be described as follows:

GLMMLasso algorithm

- Choose a starting value $(\beta^0, \phi^0, n^0)$.
- Repeat for $s = 1, 2, \ldots$
- (1) Fixed-effects parameter optimization

For $k = 1, \ldots, p$

a) (Laplace approximation)

Calculate the Laplace approximation $Q_{\lambda}(\beta, \phi)$ based on the current parameter estimates.

b) (Quadratic approximation and inexact line search)

i) Approximate the second derivative by $h_{\beta}^{(s)} > 0$.

ii) Calculate the descent direction $d_{\beta}^{(s)} \in \mathbb{R}$

iii) Choose a step size $\alpha_{\beta}^{(s)} > 0$ such that there is a decrease in the objective function.

(2) Covariance parameter optimization

For $k = 1, \ldots, d$

$\hat{\phi}_k = \arg \min_{\phi} Q_{\lambda}(\beta, \phi)$

(3) Dispersion parameter optimization

$\hat{\phi}_d = \arg \min_{\phi} Q_{\lambda}(\beta, \phi)$

until convergence.

The above algorithm solves (2) exactly. In order to speed up the algorithm, we suggest an approximate algorithm comprising the following two parts:

- Refit $u$ as fixed for the quadratic approximation with respect to the derivatives of $\beta_1$ in (1).
- Active-set algorithm: cycle through the non-zero coefficients $\beta^*_{\beta}$, and through all $p$ coefficients only every 10th iteration.

4. Two-stage GLMMLasso estimator(s)

Apart from good variable selection properties accomplished by the Lasso, we advocate a two-stage procedure to get accurate parameter estimates. Hence the first stage aims at estimating a candidate set of variables (variable screening). The goal of the second step is unbiased parameter estimation (parameter estimation).

Therefore we propose a refitting by ML methods. This two-stage procedure can be summarized as follows:

Two-stage GLMMLasso

Stage 1: Compute the GLMMLasso estimator (2).

Stage 2: Perform a ML method as in (3) or (4).

Let $(\beta_{out}, \theta_{out}, \phi_{out})$ denote the estimate from (2).

The GLMMLasso-MLE hybrid estimator

Define $\hat{b}_{out}(\beta_{out}, \theta_{out}) = \arg \min_{\beta, \phi} -\log L(\beta_{out}, \theta_{out}, \phi)$ (3)

where $S_{\lambda}(\beta, \phi) = \log L(\beta, \phi, \theta)$ and $S_{\lambda}(\beta, \phi) = \log L(\beta_{out}, \theta_{out}, \phi)$.

The thresholded GLMMLasso estimator

The thresholded Lasso with refitting was examined in Geer et al. (2010). Define $\hat{b}_{thresh} = \{ k : \hat{b}_{out}(k) \neq 0 \}$.

Let $\hat{b}_{out}(\beta_{out}, \theta_{out}) = \arg \min_{\beta, \phi} -\log L(\beta_{out}, \theta_{out}, \phi)$ (4)

Selection of the regularization parameters

For the choice of the regularization parameters $\lambda$ and $\lambda_{out}$, we propose the BIC and the AIC $c_{\lambda} = -\log L(\beta, \phi, \theta) + a(n) \cdot \| \beta \|_1$

where $a(n) = \log(n)$ for the BIC, $a(n) = 2$ for the AIC and $\| \beta \|_1 = \sum |\beta_i|$.

6. Illustration

Figure 1: Minus twice out-of-sample log-likelihood for a growing number of covariates. The MLE performs badly whereas the GLMMLasso estimators remain stable. We use a random-intercept logistic mixed model with $n = 400$, $N = 40$, $n_g = 10$, $\theta_0 = 1$, $b_0 = (0, -1, -1, -1)$.

6. R package

An implementation of the algorithm will be available online in the R package glmmlasso. The Gaussian case is implemented in the standalone R package lmmlasso (Schelldorfer et al, 2011), which is available from CRAN.

References


Jürg Schelldorfer and Peter Bühlmann

ETH Zürich

Mathematical Institute

Zürich, Switzerland