Package ‘glmmlasso’

December 22, 2011

Type Package
Title Generalized Linear Mixed Models with Lasso
Version 0.1-1
Date 2011-09-13
Author Juerg Schelldorfer
Maintainer Lukas Meier <meier@stat.math.ethz.ch>
Description This package fits generalized linear mixed models for high-dimensional data (n<<p) using a Lasso-type approach for the fixed-effects parameter.
Depends methods, Matrix(>= 0.9996875-1), lme4, glmnet
License GPL
LazyLoad yes

R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>glmmlasso-package</td>
<td>1</td>
</tr>
<tr>
<td>glmmlasso</td>
<td>2</td>
</tr>
<tr>
<td>glmmlassoControl</td>
<td>5</td>
</tr>
<tr>
<td>plot.glmmlasso</td>
<td>7</td>
</tr>
<tr>
<td>print.glmmlasso</td>
<td>8</td>
</tr>
<tr>
<td>summary.glmmlasso</td>
<td>8</td>
</tr>
<tr>
<td>xerop</td>
<td>9</td>
</tr>
</tbody>
</table>

Index 10

---

description

Fits generalized linear mixed models with a Lasso penalty for the fixed effects.

details
Package: glmmlasso
Type: Package
Version: 0.1-1
Date: 2011-09-13
License: GPL
LazyLoad: yes

This is the first version of the package and subject to testing.

Author(s)
Juerg Schelldorfer
Maintainer: Juerg Schelldorfer <schell@stat.math.ethz.ch>

---

**glmmlasso**

*Function to fit high-dimensional generalized linear mixed models.*

**Description**

Fits the solution for a high-dimensional generalized linear mixed model.

**Usage**

```r
glmlasso(y,...)
```

---

**Arguments**

- `y` response variable of length n.
- `group` grouping variable of length n.
- `X` fixed-effects matrix as an n x p matrix. An intercept has to be included in the first column as (1,...,1).
- `Z` random-effects matrix as an n x q matrix, must be in sparse Matrix format (see package Matrix).
- `family` GLM family. Currently only "binomial" and "poisson" are implemented.
- `covStruct` Covariance Structure to be used. "Identity" fits $\Sigma_\theta = \theta^2 I_q$, i.e. one single covariance parameter for all random effects. "Diagonal" fits a diagonal matrix for $\Sigma_\theta$, i.e. for each random effect a different covariance parameter.
- `lambda` non-negative regularization parameter
- `weights` weights for the fixed-effects covariates: NA means no penalization. 0 means drop this covariate; if given, the argument unpenalized is ignored. By default each covariate has weight 1.
glmlasso

coefInit starting values. They must be of the same length as the model to be fitted, i.e.
the number of variables p, the number of random-effects q and the number of
covariance parameters stot must coincide. Otherwise a warning is issued.

exactStep logical. Should the Armijo step include the update of the random effects u to
ensure that the objective function strictly decreases?

exactDeriv logical. Should the exact derivate be calculated or the derivative for fixed ran-
dom effects u.

random expression for the random-effects structure for non-correlated random effects of
the form "(1|group)+(0+X2|group)+(0+X3|group)". It is used only for generating
the corresponding Z matrix, and it dominates the Z matrix, i.e. if random
and a Z matrix is given, the Z matrix corresponding to random is used.

unpenalized indices as subset of \{1, ..., p\} indicating which fixed-effects covariates are not
subject to the $\ell_1$-penalty. Ignored if weights is given.

ranInd indices of the random effects as subset of \{1, ..., p\}. Only used for summary.glmlasso.
They need to be specified if the random-effects covariates do not correspond to
the first columns in X.

ccontrol control parameters for the algorithm, see lmmlassoControl for the details
...
not used.

Details

All the details of the algorithm can be found in the article.

Concerning logLik, deviance, aic and bic, we have to be very careful when comparing them with
other generalized linear mixed model functions. If we study a low-dimensional data example and
set $\lambda = 0$, the log-likelihood is calculated as given in the paper. Deviance, aic and bic are computed
such that they coincide with the results from glmer from the lme4 package. The latter does not
employ the standard definitions of deviance and log-likelihood function value. Nevertheless, the
differences only depends on constants, and not the parameters.

Value

A glmlasso object is returned, for which coef, resid, fitted, logLik print, summary, plot
methods exist.

fixef fixed-effects parameter beta
coefficients fixed-effects parameter beta
theta covariance parameter estimates
ranef random effects in sparse vector format
u random effects in dense vector format
objective Value of the objective function corresponding to the estimates
logLik value of the log-likelihood function. See details.
deviance value of the deviance function. See details.
aic AIC. See details.
bic BIC. See details.
activeSet Indices of the non-zero fixed-effects coefficients.
etarget The linear predictor at the current values.
mu The conditional expectation of the response at the current values.
fitted

The fitted values at the current values.

lambda

non-negative regularization parameter

weights

weights (possible adapted to the argument weights)

data

data. List with y, group, X and Z

family

GLM family used.

ntot

total number of observations

p

number of fixed-effects parameters

N

number of groups/clusters

unpenalized

indices of the non-penalized fixed-effects covariates

ranInd

indices of the random effects as subset of \{1, ..., p\}.

exactStep

logical. If the Armijo step includes the update of the random effects u or not.

exactDeriv

logical. If the exact derivate has been calculated or if the derivative for fixed random effects u has been calculated.

coefInit

starting values used in the algorithm for beta, theta and u

coefOut

list with estimates in the form required for the argument coefInit

convergence

integer giving convergence information. Each time maxArmijo was reached, convergence is increased by 2. If maxIter was reached, convergence is increased by 1.

nIter

number of outer iterations.

nIterPirls

number of pirls evaluation within the outer iteration. Pirls-Evaluation within the Armijo steps are not counted.

nfctEval

number of function evaluation. See value fctEval.

fctEval

vector of all function values calculated during the algorithm. It may be interesting if studying the convergence behaviour of the algorithm. Only if argument fctSave=TRUE

gradient

gradient of the objective function with respect to the fixed-effects coefficients

maxGrad

maximal value of the gradient which has to be close to zero for convergence.

maxArmijo

the maximal value of I used in each fixed-effects component.

ccontrol

see lmmlassoControl

resid


pearResid


respResid


workResid


devResid


Var

conditional variance of the response at the current values. See McCullagh and Nelder (1989).

di

contribution of each observation to the deviance. See McCullagh and Nelder (1989).

gof

goodness-of-fit criterion. For family="binomial", it is the in-sample misclassification rate. For family="poisson", it is the Pearson X2 statistic. See McCullagh and Nelder (1989).

cpu

cpu time needed for the algorithm.
Author(s)
Juerg Schelldorfer, <schell@stat.math.ethz.ch>

References

Examples

# (1) Use glmmlasso on the xerop data set
data(xerop)
fit1 <- glmmlasso(y=xerop$y, group=xerop$group, X=xerop$X, Z=xerop$Z,
family="binomial", covStruct="Identity", lambda=30)
summary(fit1)
plot(fit1)

# (2) Use the glmmlasso on a small simulated data set
set.seed(142)
N <- 40 # number of groups
p <- 6 # number of covariates (including intercept)
q <- 2 # number of random effect covariates
ni <- rep(10, N) # observations per group
ntot <- sum(ni) # total number of observations

group <- factor(rep(1:N, times=ni)) # grouping variable
beta <- c(0, 1, -1, 1, -1, 0) # fixed-effects coefficients
X <- cbind(1, matrix(rnorm(ntot*p), nrow=ntot)) # fixed-effects design matrix
Z <- t(glmer(rbinom(ntot, 1, runif(ntot, 0.3, 1)) ~ 1 | group) + (0 + X2 | group),
   data=data.frame(X), family="binomial")@Zt) # random-effects design matrix
bi <- c(rnorm(N, 2), rnorm(N, 1))

eta <- X%*%beta + Z%*%bi
mu <- exp(eta)/(1+exp(eta))
y <- rbinom(ntot, 1, mu@x)

# correct random effects structure
fit2 <- glmmlasso(y=y, group=group, X=X, Z=Z, family="binomial", lambda=50,
covStruct="Diagonal")
summary(fit2)
plot(fit2)

# wrong random effects structure
fit3 <- glmmlasso(y=y, group=group, X=X, family="binomial", lambda=50,
covStruct="Diagonal", random="(1 | group) + (0 + X2 | group) + (0 + X3 | group)"")
summary(fit3)
plot(fit3)
Description

Definition of various kinds of options in the algorithm.

Usage

```r
glmlassoControl(family, verbose = 0, maxIter = 200, number = 0, 
CovOpt = c("nlminb"), fctSave = TRUE, a_init = 1, delta = 0.5, 
rho = 0.1, gamm = 0, lower = 10^(-6), 
upper = ifelse(family == "binomial", 10^5, 10^3), seed = 418, 
maxArmijo = 20, min.armijo = TRUE, thres = 10^(-4), 
tol1 = 10^(-6), tol2 = 10^(-6), tol3 = 10^(-3), tol4 = 10^(-8), 
gradTol = 10^(-3))
```

Arguments

- `family` a GLM family. Currently implemented are "binomial" (default) and "poisson".
- `verbose` integer. 0 prints no output, 1 prints the outer iteration step, 2 prints the current function value, 3 prints the values of the convergence criteria
- `maxIter` maximum number of (outer) iterations
- `number` integer. Determines the active set algorithm. The zero fixed-effects coefficients are only updated each number iteration. Use 0 ≤ number ≤ 10.
- `CovOpt` character string indicating which covariance parameter optimizer to use. Currently, only "nlminb" is implemented
- `fctSave` Should all evaluation of the objective function be stored? It may help to identify the convergence pattern of the algorithm.
- `a_init` α_init in the Armijo step.
- `delta` δ in the Armijo step.
- `rho` ρ in the Armijo step.
- `gamm` γ in the Armijo step.
- `lower` lower bound for the Hessian
- `upper` upper bound for the Hessian
- `seed` set.seed in order to choose the same starting value in the cross-validation for the fixed effects
- `maxArmijo` maximum number of steps to be chosen in the Armijo step. If the maximum is reached, the algorithm continues with optimizing the next coordinate.
- `min.armijo` logical. If TRUE, the smallest l in the Armijo step is increased, as suggested in Tseng and Yun (2009). Otherwise l always starts with 0.
- `thres` if a variance or covariance parameter has smaller absolute value than thres, the parameter is set to exactly zero,
- `tol1` convergence tolerance for the relative change in the function value
- `tol2` convergence tolerance for the relative change in the fixed-effects parameters
- `tol3` convergence tolerance for the relative change in the covariance parameters
- `tol4` convergence tolerance in the PIRLS algorithm
- `gradTol` the tolerance for the gradient accepted without giving a warning
Details

For the Armijo step parameters, see Bertsekas (2003).

References


plot.glmmlasso

Diagnostic Plots for a lmlasso object

Description

Plots four diagnostic plots for checking the model assumptions and supporting model selection for a glmmlasso object.

Usage

```r
## S3 method for class 'glmmlasso'
plot(x, ...)
```

Arguments

- `x`: a lmlasso object
- `...`: not used.

Details

plot.glmmlasso shows four diagnostic plots which support checking the model assumption, model fit and may give hints for another model. 1) The first plot depicts the Tukey-Anscombe plot on the predictor scale. Points with the same color belong to the same group. 2) Plot depending on the GLM family. For family="poisson", the fitted values against the observed values is shown. For family="binomial", the Tukey-Anscombe plot on the response scale is depicted. 3) QQ-Plot of the predicted random effects. Be careful with the interpretation since the random effects have not been standardized. The color shows which points belong to the same random-effects covariate. 4) A histogram of the fixed-effects coefficients. For the interpretation of the Tukey-Anscombe plot in GLMs, see Faraway (2006).

References


Examples

```r
data(xerop)
fit <- glmmlasso(y=xerop$y, group=xerop$group, X=xerop$X, Z=xerop$Z,
family="binomial", covStruct="Identity", lambda=30)
plot(fit)
```
print.glmllasso  

Print a short summary of a glmllasso object.

Description

Prints a short summary of a glmllasso object comprising information about the variance components parameters and the number of nonzero fixed-effects coefficients.

Usage

## S3 method for class 'glmllasso'
print(x, ...)

Arguments

x  
a glmllasso object

...  
not used

See Also

summary.glmllasso

Examples

data(xerop)
fit <- glmllasso(y=xerop$y,group=xerop$group,X=xerop$X,Z=xerop$Z,  
  family="binomial",covStruct="Identity",lambda=30)
print(fit)

summary.glmllasso  

Summarize a glmllasso object

Description

Providing an elaborate summary of a glmllasso object.

Usage

## S3 method for class 'glmllasso'
summary(object, ...)

Arguments

object  
a glmllasso object

...  
not used.

Details

This functions shows a detailed summary of a glmllasso object. In the fixed-effects part, (n) right from a fixed-effects coefficient means that this coefficient was not subject to penalization.
Examples

data(xerop)
fit <- glmmlasso(y=xerop$y, group=xerop$group, X=xerop$X, Z=xerop$Z,
    family="binomial", covStruct="Identity", lambda=3)
summary(fit)

xerop

Dataset of Xerophthalmia

Description

This is a subset of the Xerophthalmia data described in Diggle et al. (2002) and Zeger and Karim (1991).

Usage

data(xerop)

Format

A list with the following four components.

Binary response variable. If the child suffers from xerophthalmia.

grp Grouping variable comprising the child id.

X Fixed-effect design matrix. The first column is the intercept, then age, xero, cos, sin, sex, height and stunt. The covariates are all standardized with mean 0 and variance 1.

Z Random-effects design matrix for a random-intercept model.

Details

A detailed description of the covariates can be found in Diggle et al. and Zeger and Karim (1991).

Source

http://faculty.washington.edu/heagerty/Books/AnalysisLongitudinal/xerop.data

References


Examples

data(xerop)
Index

* Topic **datasets**
  xerop, 9

* Topic **hplot**
  plot.glmmlasso, 7

* Topic **methods**
  print.glmmlasso, 8
  summary.glmmlasso, 8

* Topic **misc**
  glmmlassoControl, 5

* Topic **models**
  glmmlasso, 2

* Topic **package**
  glmmlasso-package, 1

* Topic **regression**
  glmmlasso, 2
  glmmlasso-package, 1
  glmmlassoControl, 5

  plot.glmmlasso, 7
  print.glmmlasso, 8

  summary.glmmlasso, 8
  xerop, 9