

Package ‘TrioSGL’

October 12, 2022

Type Package

Encoding UTF-8

Title Trio Model with a Combination of Lasso and Group Lasso
Regularization

Version 1.1.0

Date 2017-12-18

Author Timo Stöcker, Noah Simon, Jerome Friedman, Trevor Hastie, and Rob Tibshirani

Maintainer Timo Stöcker <timo.stoecker@hhu.de>

Description Fit a trio model via penalized maximum likelihood. The model is fit for a path of values of the penalty parameter. This package is based on Noah Simon, et al. (2011) <doi:10.1080/10618600.2012.681250>.

NeedsCompilation yes

License GPL

Repository CRAN

Date/Publication 2017-12-21 09:40:55 UTC

R topics documented:

TrioSGL-package	1
TrioSGL	3
Index	5

TrioSGL-package	<i>Trio Model with a Combination of Lasso and Group Lasso Regularization</i>
-----------------	--

Description

Fit a trio model via penalized maximum likelihood. The model is fit for a path of values of the penalty parameter. This package is based on Noah Simon, et al. (2011) <doi:10.1080/10618600.2012.681250>.

Details

Package: TrioSGL
 Type: Package
 Version: 1.1.0
 Date: 2017-12-18
 License: GPL
 LazyLoad: yes

Only 1 function: TrioSGL

Author(s)

Timo Stöcker, Noah Simon, Jerome Friedman, Trevor Hastie, and Rob Tibshirani
 Maintainer: Timo Stöcker <timo.stoecker@hhu.de>

References

Simon, N., Friedman, J., Hastie, T., Tibshirani, R. (2011). *A Sparse-Group Lasso*. *Journal of Computational and Graphical Statistics*, 22, 231-245.

TrioSGL	<i>Trio Model with a Combination of Lasso and Group Lasso Regularization</i>
---------	--

Description

Fit a trio model via penalized maximum likelihood. The model is fit for a path of values of the penalty parameter. This package is based on Noah Simon, et al. (2011) <doi:10.1080/10618600.2012.681250>.

Usage

```
TrioSGL(X, index = NULL, maxit = 10000, thresh = 0.0001, min.frac = 0.01,
        nlam = 25, lambdas = NULL, alpha = 0.95, gamma = 0.8, step = 1, reset = 20,
        standardize = FALSE, verbose = FALSE)
```

Arguments

X	\$X\$ should be an input matrix of dimension n-obs by p-vars. The number of rows must be a multiple of 4 (case followed by 3 pseudo-controls).
index	A p-vector indicating group membership of each covariate
maxit	Maximum number of iterations to convergence
thresh	Convergence threshold for change in beta
min.frac	The minimum value of the penalty parameter, as a fraction of the maximum value
nlam	Number of lambda to use in the regularization path

lambdas	A user specified sequence of lambda values for fitting. We recommend leaving this NULL and letting TrioSGL self-select values
alpha	The mixing parameter. $\alpha = 1$ is the lasso penalty. $\alpha = 0$ is the group lasso penalty.
gamma	Fitting parameter used for tuning backtracking (between 0 and 1)
step	Fitting parameter used for initial backtracking step size (between 0 and 1)
reset	Fitting parameter used for taking advantage of local strong convexity in nesterov momentum (number of iterations before momentum term is reset)
standardize	Logical flag for variable standardization prior to fitting the model.
verbose	Logical flag for whether or not step number will be output

Details

The sequence of models along the regularization path is fit by accelerated generalized gradient descent.

Value

An object with S3 class "TrioSGL"

beta	A p by n_{lam} matrix, giving the penalized MLEs for the n_{lam} different models, where the index corresponds to the penalty parameter λ
lambdas	The actual sequence of lambda values used (penalty parameter)
X.transform	A list used in predict which gives the empirical mean and variance of the x matrix used to build the model

Author(s)

Timo Stöcker, Noah Simon, Jerome Friedman, Trevor Hastie, and Rob Tibshirani
 Maintainer: Timo Stöcker <timo.stoecker@hhu.de>

References

Simon, N., Friedman, J., Hastie, T., Tibshirani, R. (2011). *A Sparse-Group Lasso*. Journal of Computational and Graphical Statistics, 22, 231-245.

Examples

```
trios <- 4 * 10; snps <- 20; size.groups <- 4
index <- ceiling(1:snps / size.groups)
x <- floor(matrix(runif(trios * snps, min = 0, max = 3), ncol = snps, nrow = trios))
fit <- TrioSGL(x, index)
```

Index

* **model**

TrioSGL, [3](#)

* **regression**

TrioSGL, [3](#)

* **trio**

TrioSGL, [3](#)

TrioSGL, [3](#)

TrioSGL-package, [1](#)