Package ‘higlasso’

May 25, 2020

Title Hierarchical Integrative Group LASSO

Version 0.9.0

Description Environmental health studies are increasingly measuring multiple pollutants to characterize the joint health effects attributable to exposure mixtures. However, the underlying dose-response relationship between toxicants and health outcomes of interest may be highly nonlinear, with possible nonlinear interaction effects. Hierarchical integrative group least absolute shrinkage and selection operator (HiGLASSO), developed by Boss et al (2020) <arXiv:2003.12844>, is a general framework to identify noteworthy nonlinear main and interaction effects in the presence of group structures among a set of exposures.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Depends R (>= 3.5.0)

Imports gcdnet, gglasso, purrr, splines, Rcpp

LinkingTo Rcpp, RcppArmadillo

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

NeedsCompilation yes

Author Alexander Rix [aut, cre],
Jonathan Boss [aut]

Maintainer Alexander Rix <alexrix@umich.edu>

Repository CRAN

Date/Publication 2020-05-25 17:40:03 UTC

R topics documented:

cv.higlasso 2

higlasso 4
higlasso.df 6
print.cv.higlasso 7

Index **8**

`cv.higlasso` *Cross Validated Hierarchical Integrative Group LASSO*

Description

Does k-fold cross-validation for `higlasso`, and returns optimal values for `lambda1` and `lambda2`.

Usage

```
cv.higlasso(
  Y,
  X,
  Z,
  method = c("aenet", "gglasso"),
  lambda1 = NULL,
  lambda2 = NULL,
  nlambda1 = 10,
  nlambda2 = 10,
  lambda.min.ratio = 0.05,
  nfolds = 5,
  foldid = NULL,
  sigma = 1,
  degree = 2,
  maxit = 5000,
  tol = 1e-05
)
```

Arguments

`Y`	A length n numeric response vector
`X`	A n x p numeric matrix
`Z`	A n x m numeric matrix
`method`	Type of initialization to use. Possible choices are `gglasso` for group LASSO and `aenet` for adaptive elastic net. Default is `aenet`
`lambda1`	A numeric vector of main effect penalties on which to tune By default, `lambda1 = NULL` and higlasso generates a length `nlambda1` sequence of lambda1s based off of the data and `min.lambda.ratio`
`lambda2`	A numeric vector of interaction effects penalties on which to tune. By default, `lambda2 = NULL` and generates a sequence (length `nlambda2`) of lambda2s based off of the data and `min.lambda.ratio`

`nlambda1` The number of lambda1 values to generate. Default is 10, minimum is 2. If `lambda1 != NULL`, this parameter is ignored

`nlambda2` The number of lambda2 values to generate. Default is 10, minimum is 2. If `lambda2 != NULL`, this parameter is ignored

`lambda.min.ratio` Ratio that calculates min lambda from max lambda. Ignored if 'lambda1' or 'lambda2' is non NULL. Default is 0.05

`nfolds` Number of folds for cross validation. Default is 10. The minimum is 3, and while the maximum is the number of observations (ie leave one out cross validation)

`foldid` An optional vector of values between 1 and `max(foldid)` identifying what fold each observation is in. Default is NULL and `cv.higlasso` will automatically generate `foldid` based off of `nfolds`

`sigma` Scale parameter for integrative weights. Technically a third tuning parameter but defaults to 1 for computational tractability

`degree` Degree of `bs` basis expansion. Default is 2

`maxit` Maximum number of iterations. Default is 5000

`tol` Tolerance for convergence. Defaults to 1e-5

Details

There are a few things to keep in mind when using `cv.higlasso`

- `higlasso` uses the strong heredity principle. That is, `X_1` and `X_2` must included as main effects before the interaction `X_1 X_2` can be included.
- While `higlasso` uses integrative weights to help with estimation, `higlasso` is more of a selection method. As a result, `cv.higlasso` does not output coefficient estimates, only which variables are selected.
- Simulation studies suggest that `higlasso` is a very conservative method when it comes to selecting interactions. That is, `higlasso` has a low false positive rate and the identification of a nonlinear interaction is a good indicator that further investigation is worthwhile.
- `cv.higlasso` can be slow, so it may may be beneficial to tweak some of its settings (for example, `nlambda1`, `nlambda2`, and `nfolds`) to get a handle on how long the method will take before running the full model.

As a side effect of the conservativeness of the method, we have found that using the 1 standard error rule results in overly sparse models, and that `lambda.min` generally performs better.

Value

An object of type `cv.higlasso` with 7 elements

lambda An `nlambda1` x `nlambda2` x 2 array containing each pair `(lambda1,lambda2)` pair.

lambda.min lambda pair with the lowest cross validation error

lambda.1se

cvm cross validation error at each lambda pair. The error is calculated from the mean square error.

cvse standard error of cvm at each lambda pair.

higlasso.fit higlasso output from fitting the whole data.

call The call that generated the output.

Author(s)

Alexander Rix

References

A Hierarchical Integrative Group LASSO (HiGLASSO) Framework for Analyzing Environmental Mixtures. Jonathan Boss, Alexander Rix, Yin-Hsiu Chen, Naveen N. Narisetty, Zhenke Wu, Kelly K. Ferguson, Thomas F. McElrath, John D. Meeker, Bhramar Mukherjee. 2020. arXiv:2003.12844

Examples

```
library(higlasso)

X <- as.matrix(higlasso.df[, paste0("V", 1:7)])
Y <- higlasso.df$Y
Z <- matrix(1, nrow(X))


# This can take a bit of time

fit <- cv.higlasso(Y, X, Z)

print(fit)
```

higlasso *Hierarchical Integrative Group LASSO*

Description

HiGLASSO is a regularization based selection method designed to detect non-linear interactions between variables, particularly exposures in environmental health studies.

Usage

```
higlasso(
  Y,
  X,
  Z,
  method = c("aenet", "gglasso"),
  lambda1 = NULL,
  lambda2 = NULL,
  nlambda1 = 10,
  nlambda2 = 10,
  lambda.min.ratio = 0.05,
  sigma = 1,
  degree = 2,
  maxit = 5000,
  tol = 1e-05
)
```

Arguments

`Y` A length n numeric response vector

`X` A n x p numeric matrix of covariates to basis expand

`Z` A n x m numeric matrix of non basis expanded and non regularized covariates

`method` Type of initialization to use. Possible choices are `gglasso` for group LASSO and `aenet` for adaptive elastic net. Default is `aenet`

`lambda1` A numeric vector of main effect penalties on which to tune By default, `lambda1 = NULL` and higlasso generates a length `nlambda1` sequence of lambda1s based off of the data and `min.lambda.ratio`

`lambda2` A numeric vector of interaction effects penalties on which to tune. By default, `lambda2 = NULL` and generates a sequence (length `nlambda2`) of lambda2s based off of the data and `min.lambda.ratio`

`nlambda1` The number of lambda1 values to generate. Default is 10, minimum is 2. If `lambda1 != NULL`, this parameter is ignored

`nlambda2` The number of lambda2 values to generate. Default is 10, minimum is 2. If `lambda2 != NULL`, this parameter is ignored

`lambda.min.ratio`
Ratio that calculates min lambda from max lambda. Ignored if 'lambda1' or 'lambda2' is non NULL. Default is 0.05

`sigma` Scale parameter for integrative weights. Technically a third tuning parameter but defaults to 1 for computational tractability

`degree` Degree of `bs` basis expansion. Default is 2

`maxit` Maximum number of iterations. Default is 5000

`tol` Tolerance for convergence. Default is 1e-5

Details

There are a few things to keep in mind when using `higlasso`

- `higlasso` uses the strong heredity principle. That is, `X_1` and `X_2` must included as main effects before the interaction `X_1 X_2` can be included.
- While `higlasso` uses integrative weights to help with estimation, `higlasso` is more of a selection method. As a result, `higlasso` does not output coefficient estimates, only which variables are selected.
- Simulation studies suggest that `higlasso` is a very conservative method when it comes to selecting interactions. That is, `higlasso` has a low false positive rate and the identification of a nonlinear interaction is a good indicator that further investigation is worthwhile.

- higlasso can be slow, so it may may be beneficial to tweak some of its settings (for example, nlambda1 and nlambda2) to get a handle on how long the method will take before running the full model.

Value

An object of type "higlasso" with 4 elements:

lambda An nlambda1 x nlambda2 x 2 array containing each pair (lambda1,lambda2) pair.

selected An nlambda1 x nlambda2 x ncol(X) array containing higlasso's selections for each lambda pair.

df The number of nonzero selections for each lambda pair.

call The call that generated the output.

Author(s)

Alexander Rix

References

A Hierarchical Integrative Group LASSO (HiGLASSO) Framework for Analyzing Environmental Mixtures. Jonathan Boss, Alexander Rix, Yin-Hsiu Chen, Naveen N. Narisetty, Zhenke Wu, Kelly K. Ferguson, Thomas F. McElrath, John D. Meeker, Bhramar Mukherjee. 2020. arXiv:2003.12844

Examples

```
library(higlasso)

X <- as.matrix(higlasso.df[, paste0("V", 1:7)])
Y <- higlasso.df$Y
Z <- matrix(1, nrow(X))


# This can take a bit of time
higlasso.fit <- higlasso(Y, X, Z)
```

---

higlasso.df *Synthetic Example Data For Higlasso*

Description

This synthetic data is taken from the linear interaction simulations from the higlasso paper. The data generating model is:

$$Y = X_1 + X_2 + X_3 + X_4 + X_5 + X_1X_2 + X_1X_3 + X_2X_3$$

$$+X_1X_4 + X_2X_4 + X_3X_4 + X_1X_5$$

$$+X_2X_5 + X_3X_5 + X_4X_5 + \epsilon$$

Usage

```
higlasso.df
```

Format

A data.frame with 500 observations on 11 variables:

Y Continuous response.

X1-X10 Covariates.

`print.cv.higlasso` *Print CV HiGLASSO Objects*

Description

`print.cv.higlasso` prints a fitted "cv.higlaso" object and returns it invisibly.

Usage

```
## S3 method for class 'cv.higlasso'
print(x, ...)
```

Arguments

`x`	An object of type "cv.higlasso" to print
`...`	Further arguments passed to or from other methods

Value

The original input, `x` (invisibly).

Index

∗Topic **datasets**
    higlasso.df, 6

cv.higlasso, 2

higlasso, 4
higlasso.df, 6

print.cv.higlasso, 7