

Package ‘ridgeAdap’

June 2, 2016

Title Adaptive Ridge for variable selection

Version 0.3

Date 2016-05-31

Author Jean-Michel Becu

Maintainer Jean-Michel Becu <jean-michel.becu@hds.utc.fr>

Depends R (>= 2.15),quadrupen,SGL

Description ridgeAdap

License GPL

Repository CRAN

R topics documented:

ridgeAdap-package	1
cleaning.lm	4
cleaning.ridge	5
ridge	6
ridge.permut	7
screening	8
screening.group	9
Index	11

ridgeAdap-package	<i>Adaptive ridge</i>
-------------------	-----------------------

Description

This package is dedicated to feature selection and estimation based on the adaptive ridge method.

Details

The key idea of the adaptive ridge regression is the use of specific penalty into the ridge regression that could mimic lasso and it's variation. The equivalence between ridge regression with specific penalty and lasso show that this penalty is the better way to transfert signal from lasso to ridge.

The adaptive ridge constitute same at the adaptive lasso, to learn elastic-net or sparse group lasso estimates in a first step, and use the estimates of lasso to define a specific penalty to the ridge

regression. This penalty definition differs if elastic-net or sparse group lasso was used in the first step, that explain the name of "adaptive" ridge regression.

Two application are possible :

Variable selection : Two step protocol based on Larry Wasserman (2009) works.

Af first step ([screening](#), [screening.group](#)), to select variables by a sparse approach on the half of the dataset. This package use the Elastic-net regression, which is write as follows :

$$\hat{\beta}_{\lambda_1, \lambda_2} = \arg \min_{\beta} RSS(beta) + \lambda_1 \|\beta\|_1 + \lambda_2 \|\beta\|_2.$$

A second step ([cleaning.ridge](#)), to test variables in the subset of variables selected in the first step. This step is applied on the second half of the dataset with the adaptive ridge regression, which is write as follows :

$$\hat{\beta}_{\Omega} = \arg \min_{\beta} RSS(beta) + \sum \omega_j \beta_j,$$

with a specific penalty for the elastic-net

$$\omega_j = \frac{\lambda_1}{|\hat{\beta}_j(\lambda_1)|} + \lambda_2.$$

The significance test for the ridge regressor is based on the fisher's staistic and permutations.

Parameters λ_1, λ are chosen by k-folds Cross-Validation on the entire process.

This package is under development, all feedback are appreciate.

Author(s)

Jean-Michel Becu.

Maintener : Jean-Michel Becu (jean-michel.becu@inria.fr)

References

Becu J.M., Ambroise C., Dalmaso C. and Grandvalet Y. Beyond Support in Two-Stage Variable Selection. *Statistics and Computing*, **26**:1-11,2016.

For further information :

L. Wasserman and K. Roeder. High-dimensional variable selection. *The Annals of Statistics*, **37**(5A):2178-2201, 2009. Wasserman

Y. Grandvalet and S. Canu. Outcomes of the equivalence of adaptive ridge with least absolute shrinkage. In M. S. Kearns, S. A. Solla, and D. A. Cohn, editors, *Advances in Neural Information Processing Systems 11 (NIPS 1998)*, pages 445-451. MIT Press, 1999.

Examples

```
library("ridgeAdap")
#generate dataset with block's type
n <- 250
p <- 500
rho <- 0.5

nRel <- 25
SNR <- 4
group<-sort(rep(1:20,25))
```

```

S <-matrix(unlist(lapply(1:p,FUN=function(x){group == group[x]})),ncol=p)
sigma<-matrix(0,p,p)
sigma[S==TRUE]<-rho
diag(sigma)<-1

beta<-rep(0,p)
beta[sample(1:p,nRel)]<-runif(nRel,0.1,1)
U <- t(chol(sigma))
random.normal <- matrix(rnorm(p*n,0,1), nrow=p,ncol=n)
X <- U%% random.normal
X<-t(X)

y<-as.vector(t(matrix(beta,nrow=p))%%t(X))
var<-sqrt(t(matrix(beta,ncol=1))%%sigma%%matrix(beta,ncol=1))*(1/(SNR))
y<-y+rnorm(n,0,var)

##selection procedure
subsets <-split(1:n,1:2)
Screen<-screening(X[subsets[[1]],],y[subsets[[1]]])
Clean<-cleaning.ridge(X[subsets[[2]],],y[subsets[[2]]],screening=Screen)
selectVar<-as.integer(names(which(p.adjust(Clean$pval,"BH") <= 0.05)))

Screen.group<-screening.group(X[subsets[[1]],],y[subsets[[1]]],group=group)
Clean.group<-cleaning.ridge(X[subsets[[2]],],y[subsets[[2]]],screening=Screen.group,group=group)

selectGroup<-as.integer(names(which(p.adjust(Clean.group$pval,"BH") <= 0.05)))

real <- which(beta!=0) ## index for variables with beta != 0
realInGroup <- table(group[beta!=0]) ## Number of variables with beta!=
0 for explanatory groups

##estimation procedure

n <- 250
p <- 500
rho <- 0.5

nRel <- 25
SNR <- 4
group<-sort(rep(1:20,25))

S <-matrix(unlist(lapply(1:p,FUN=function(x){group ==
group[x]})),ncol=p)
sigma<-matrix(0,p,p)
sigma[S==TRUE]<-rho
diag(sigma)<-1

beta<-rep(0,p)
beta[sample(1:p,nRel)]<-runif(nRel,0.1,1)
U <- t(chol(sigma))
random.normal <- matrix(rnorm(p*n,0,1), nrow=p,ncol=n)
X <- U%% random.normal
X<-t(X)

random.normal <- matrix(rnorm(p*n,0,1), nrow=p,ncol=n)
X.test <- U%% random.normal

```

```

X.test<-t(X.test)

y<-as.vector(t(matrix(beta,nrow=p))%*%t(X))
var<-sqrt(t(matrix(beta,ncol=1))%*%sigma%*%matrix(beta,ncol=1))*(1/(SNR))
y<-y+rnorm(n,0,var)

esti<-estimation.AR(X,y)

esti$beta.min

```

cleaning.lm

Cleaning.lm

Description

Cleaning.lm

Usage

```

cleaning.lm(X,y,screening=NULL,beta=rep(1,ncol(X)),test="student",
group=1:ncol(X),scale=FALSE,center=TRUE)

```

Arguments

X	a matrix object with n*p size representing the data
y	a vector of size p representing the output variable
screening	object issued from the screening or screening.group functions. If is not NULL then the beta.min vector contained in this object will be used instead the beta main parameter.
beta	a vector of size p representing the weight of variable. A variable with weight equal to zero signify will not appear in the ordinary least square regression. The number of variables with a non-zero weight must be inferior to the sample size of X.
test	a string of characters to determine the statistical test to be used. This should be one of "student" or "fisher". Student or fisher test are the same when variables are not grouped. If variables are grouped then the student t-test measure the significancy of each variables and the fisher F-test measure the significancy of each groups of variables. Default is "student".
group	a vector of size p representing the group index for each variables. Default is 1:ncol(X) which represent the particular case whit no group. Warning : variables and this vector must be in ascending group order.
scale	a boolean to indicate if X and y will be scaled. Default is FALSE.
center	a boolean to indicate if X and y will be centered. Default is TRUE.

Value

An object of class 'cleaning' with pvalue for each variables or group variables corresponding to the statistical test choice. Contain equally estimates of ols and t or F statistics.

Author(s)

JM BECU

Examples

```
# see ridgeAdap help
```

cleaning.ridge

*Cleaning.ridge***Description**

Cleaning.ridge

Usage

```
cleaning.ridge(X,y,screening=NULL,lambda1=0,lambda2=0,alpha,beta=rep(1,ncol(X)),
nbIter=1000,group=1:ncol(X),penalty="quadrupen",scale=FALSE,center=TRUE)
```

Arguments

X	a matrix object with n*p size representing the data
y	a vector of size p representing the output variable
screening	object issued from the screening or screening.group functions. If is not NULL then the beta.min, lambda1.min, lambda2.min, penalty type informations contained in this object will be used instead the beta main parameter. Default is NULL.
lambda1	The l1 penalty to used.
lambda2	The l2 penalty to use in case of elatic-net for screening.
alpha	The α mixing parameter to use in case of sparse group lasso for screening or if glmnet is used for screening.
beta	a vector of size p representing the weight of variable. A variable with weight equal to zero signify will not appear in the ordinary least square regression. No constraint to the number of variables with a non-zero weight.
nbIter	Number of iterations used to simulate the null hypothesis law. Bigger is better. Default is 1000. Could be less with the gamma law approximation.
group	a vector of size p representing the group index for each variables. Default is 1:ncol(X) which represent the particular case whit no group. Warning : this vector must be in ascending group order and variables ordered in this sense.
penalty	a string of characters to determine the adaptive ridge specific penalty to be used. This should be one of "quadrupen" to elastic-net by quadrupen package, "glmnet" to elastic-net by glmnet package, "grplasso" to group lasso by grplasso package and "SGL" to sparse group lasso by SGL package at screening. By default is "quadrupen".
scale	a boolean to indicate if X and y will be scaled. Default is FALSE.
center	a boolean to indicate if X and y will be centered. Default is TRUE.
lawWithOne	a boolean to indicate if one distribution under the null hypothesis for all groups of variables could be used instead of one distribution for each of them. Warning : It is an approximation and that work only if all groups have the same size. Default is FALSE.

Value

An object of class 'cleaning' with pvalue for each variables or group variables. Two pvalues are given, the first is estimates on the real distribution of F-statistic under the null hypothesis, the second pval.gamma is estimates on the gamma low fitting on the real distribution of F-statistic. This approximation could be used to minimize the number of needed iteration. Contains equally estimates of adaptive ridge, F statistics on estimates and F statistics simulated under the H0 model.

Author(s)

JM BECU

Examples

```
# see ridgeAdap help
```

ridge	<i>Ridge</i>
-------	--------------

Description

Ridge

Usage

```
ridge(X,y,lambda1=0,lambda2=0,alpha=1,beta=rep(1,ncol(X)),group=1:ncol(X),
penalty="quadrupen")
```

Arguments

X	an matrix object with n*p size
y	an vector of size p
lambda1	The l1 penalty to used.
lambda2	The l2 penalty to use in case of elatic-net for screening.
alpha	The α mixing parameter to use in case of sparse group lasso for screening or if glmnet is used for screening.
beta	a vector of size p representing the weight of variable. A variable with weight equal to zero signify will not appear in the ordinary least square regression. No constraint to the number of variables with a non-zero weight.
group	a vector of size p representing the group index for each variables. Default is 1:ncol(X) which represent the particular case whit no group. Warning : this vector must be in ascending group order and variables ordered in this sense.
penalty	a string of characters to determine the adaptive ridge specific penalty to be used. This should be one of "quadrupen" to elastic-net by quadrupen package, "glmnet" to elastic-net by glmnet package, "grplasso" to group lasso by grplasso package and "SGL" to sparse group lasso by SGL package at screening. By default is "quadrupen".

Value

An list with ridge estimates, y estimates.

Author(s)

JM BECU

Examples

```
# see ridgeAdap help
```

ridge.permut	<i>Ridge.permut</i>
--------------	---------------------

Description

Ridge.permut

Usage

```
ridge.permut(X,y,lambda1=0,lambda2=0,alpha=1,beta=rep(1,ncol(X)),iter=1000,
group=1:ncol(X),penalty="quadrupen")
```

Arguments

X	an matrix object with n*p size
y	an vector of size p
lambda1	The l1 penalty to used.
lambda2	The l2 penalty to use in case of elatic-net for screening.
alpha	The α mixing parameter to use in case of sparse group lasso for screening or if glmnet is used for screening.
beta	a vector of size p representing the weight of variable. A variable with weight equal to zero signify will not appear in the ordinary least square regression. No constraint to the number of variables with a non-zero weight.
iter	Number of iterations used to simulate the null hypothesis law. Bigger is better. Default is 1000.
group	a vector of size p representing the group index for each variables. Default is 1:ncol(X) which represent the particular case whit no group. Warning : this vector must be in ascending group order and variables ordered in this sense.
penalty	a string of characters to determine the adaptive ridge specific penalty to be used. This should be one of "quadrupen" to elastic-net by quadrupen package, "glmnet" to elastic-net by glmnet package, "grplasso" to group lasso by grplasso package and "SGL" to sparse group lasso by SGL package at screening. By default is "quadrupen".
lawWithOne	a boolean to indicate if one distribution under the null hypothesis for all groups of variables could be used instead of one distribution for each of them. Warning : It is an approximation and that work only if all groups have the same size. Default is FALSE.

Value

An list with ridge estimates, F statistic an F statistics under H0 and p-value for each variables or group significancy. Contains equally pval.gamma obtaines with the gamma law approximation.

Author(s)

JM BECU

Examples

```
# see ridgeAdap help
```

screening	<i>Screening</i>
-----------	------------------

Description

Screening

Usage

```
screening(X,y,lambda1 = NULL,lambda2=0,nfolds=10,scale=FALSE,center=TRUE,se1=FALSE)
```

Arguments

X	a matrix object with n*p size representing the data
y	a vector of size p representing the output variable
lambda1	a vector representing the l1 penalty. Default is NULL and automatically defined by quadrupen.
lambda2	a vector representing the l2 penalty. If lambda2 = 0 that correspond to the lasso solution, otherwise to the elastic-net solution. Default is 0.
nfolds	An integer representing the number of folds for the k-folds cross-validation used to choose l1 and l2 penatly. Default is 10 and must be greater than 2.
scale	a boolean to indicate if X and y will be scaled. Default is FALSE.
center	a boolean to indicate if X and y will be centered. Default is TRUE.
se1	a boolean to indicate if using the se1 rule for the cross validation with nfolds upper than 2.

Value

A object of class 'screening' with 5 elements

beta.min	a vector of size p with the elastic.net estimates
S	a vector cantaning the index of variable which $B_j \neq 0$
lambda1.min	l1 penalty chosen by CV
lambda2.min	l2 penalty chosen by CV
ref	a list with two component contains respectively the X and y variance and mean used to scaling and/or centering the data.

Note

screening requires quadrupen package.

Author(s)

JM BECU

Examples

```
# see ridgeAdap help
```

screening.group

*Screening.group***Description**

Screening.group

Usage

```
screening.group(X,y,lambda = NULL,alpha=0,nlam=20,nfolds=10,group=1:ncol(X),
thresh=0.001,scale=FALSE,center=TRUE)
```

Arguments

X	a matrix object with n*p size representing the data
y	a vector of size p representing the output variable
lambda	a vector representing the l1 penalty. Default is NULL and automatically defined by SGL.
alpha	a vector representing the α parameter for the sparse group lasso. If alpha = 1 that correspond to the lasso solution, if alpha = 0 that correspond to the group lasso and otherwise to the sparse group lasso solution. Default is 0.
nlam	an integer representing the size of the lambda grid if lambda is NULL. Default is 20.
nfolds	an integer representing the number of folds for the k-folds cross-validation used to choose l1 and l2 penatly. Default is 10 and must be greater than 2.
group	a vector of size p representing the group index for each variables. Default is 1:ncol(X) which represent the particular case whit no group. Warning this vector must be in ascending group order and variables in X needs to be ordered in this sense.
thresh	convergence threshold for change in beta.
scale	a boolean to indicate if X and y will be scaled. Default is TRUE.
center	a boolean to indicate if X and y will be centered. Default is TRUE.

Value

A object of class 'screening' with 5 elements

beta.min	a vector of size p with the elastic.net estimates
S	a vector contaning the index of variable which $\$B_j \neq 0$
lambda1.min	l1 penalty choose by CV
alpha.min	alpha mixing parameter choose by CV
ref	a list with two component contains respectively the X and y variance and mean used to scaling and/or centering the data.

Note

screening.group requires SGL package.

Author(s)

JM BECU

Examples

```
# see ridgeAdap help
```

Index

- *Topic **cleaning**
 - cleaning.lm, [4](#)
 - cleaning.ridge, [5](#)
- *Topic **package**
 - ridgeAdap-package, [1](#)
- *Topic **ridge**
 - ridge, [6](#)
 - ridge.permut, [7](#)
- *Topic **screening**
 - screening, [8](#)
 - screening.group, [9](#)

cleaning.lm, [4](#)
cleaning.ridge, [2](#), [5](#)

ridge, [6](#)
ridge.permut, [7](#)
ridgeAdap (ridgeAdap-package), [1](#)
ridgeAdap-package, [1](#)

screening, [2](#), [8](#)
screening.group, [2](#), [9](#)