Package ‘LassoSIR’

March 31, 2025

Type Package

Title Sparsed Sliced Inverse Regression via Lasso

Version 1.0

Date 2025-03-31

Maintainer Zhigen Zhao <zhigen.zhao@gmail.com>

Description Estimate the sufficient dimension reduction space using sparsed sliced inverse regression via Lasso (Lasso-SIR) introduced in Lin, Zhao, and Liu (2019) <doi:10.1080/01621459.2018.1520115>. The Lasso-SIR is consistent and achieve the optimal convergence rate under certain sparsity conditions for the multiple index models.

License GPL-3

Imports glmnet, graphics, stats

NeedsCompilation no

Author Zhigen Zhao [aut, cre],
Qian Lin [aut],
Jun Liu [aut]

Repository CRAN

Date/Publication 2025-03-31 19:00:12 UTC

Contents

LassoSIR *LassoSIR*

Description

This function calculates the sufficient dimension reduction (SDR) space using the Sparse Sliced Inverse Regression Via Lasso (Lasso-SIR).

The input is a continuous design matrix X and a response vector Y which can be either continuous or categorical. X is arranged such that each column corresponds to one variable and each row corresponds to one subject.

The function gives users options to choose (i) the dimension of the SDR space, (ii) screening based on the diagonal thresholding, (iii) the number of slices (H), and many others.

Usage

```
LassoSIR(X, Y, H = 0, choosing.d = "automatic", solution.path = FALSE,
        categorical = FALSE, nfolds = 10, screening = TRUE, no.dim = 0)
```

Arguments

`X` This argument is the continuous design matrix X. X is arranged such that each column corresponds to one variable and each row corresponds to one subject.

`Y` The response vector Y, which can be either continuous or categorical.

`H` The number of slices.

(i) If the boolean variable "categorical" is true, H is chosen as the number of categories automatically.

(ii) If the response variable is continuous, namely, "categorical" is false, user need to specify the number of slices. If H is set as 0, the code will ask the user to enter the number of slices interactively;

(iii) the default choice of H is zero.

`choosing.d` This argument asks for the method of choosing the dimension of SDR. If no.dim is non zero, then choosing.d is set as "given". Otherwise, choosing.d can be set as "automatic" or "manual".

When choosing.d is set as "manual", this function will calculate the eigenvalues of var(EX|Y) and plot these eigenvalues. After that, the user will be asked to enter the dimension interactively.

When choosing.d is set as "automatic", the dimension will be determined automatically according to Algorithm 5 from the original paper.

The default option is "automatic".

`solution.path` When setting this boolean variable as TRUE, a plot with solution path based on the final proposed model will be plotted.

The default option is FALSE.

`categorical`	When setting this boolean variable as TRUE, the response variable is categorical; otherwise, the response variable is continuous. The default option is FALSE.
`nfolds`	This argument set the number of folds in the cross validation. The default option is 10.
`screening`	When setting this boolean variable as TRUE, a diagonal thresholding (DT-SIR) step is applied to reduce the dimension before applying Lasso-SIR.
`no.dim`	This argument specifies the dimension of SDR. The default option is 0 and this dimension is chosen manually or automatically based on the choice of choosing.d.

Details

This function estimates the sufficient dimension reduction space using the sparse sliced inverse regression for high dimensional data via Lasso (LassoSIR).

Value

When solution.path is set as true, the function returns a glmnet object.

When solution.path is set as false, the tuning parameter in Lasso is chosen by using the cross validation. The function returns the following values:

`beta`	the estimated coefficient in SDR.
`eigen.value`	the eigen value of the estimator of $var(EY\|X)$.
`no.dim`	the dimension of the central space.
`H`	the number of slices.
`categorical`	a boolean variable to indicate the type of the response.

Note

NA

Author(s)

Zhigen Zhao, Qian Lin, Jun S. Liu

References

Lin, Q., Zhao, Z. , and Liu, J. (2018) On consistency and sparsity for sliced inverse regression in high dimension. Annals of Statistics. Vol. 46, Number 2. Page 580-610.

Lin, Q., Zhao, Z. , and Liu, J. (2019) Sparse Sliced Inverse Regression for High Dimensional Data. Journal of the American Statistical Association. Vol. 114, Number 528, Page 1726-1739.

See Also

NA

Examples

```
p <- 10
n <- 200


H <- 20
m <- n/H

beta <- array(0, c(p, 1) )
beta[1:3,1] <- rnorm(3, 0, 1)

X <- array(0, c(n, p ) )

rho <- 0.3
Sigma <- diag(p)
elements <- rho^(c((p-1):0,1:(p-1) ) )
for(i in 1:p)
    Sigma[i,] <- elements[(p+1-i):(2*p-i) ]


X <- matrix( rnorm(p*n), c(n, p) )
X <- X%*% chol(Sigma)

Y <-  ( X%*% beta )^3/2 + rnorm(n,0,1)
sir.lasso <- LassoSIR( X, Y, H, choosing.d="automatic",
          solution.path=FALSE, categorical=FALSE, nfolds=10,
          screening=FALSE)
```

`LassoSIR-Package` *Sparsed Sliced Inverse Regression via Lasso*

Description

Estimate the sufficient dimension reduction space using sparsed sliced inverse regression via Lasso (Lasso-SIR) introduced in Lin, Zhao, and Liu (2019) <doi:10.1080/01621459.2018.1520115>. The Lasso-SIR is consistent and achieve the optimal convergence rate under certain sparsity conditions for the multiple index models.

Details

The DESCRIPTION file:

Package:	LassoSIR
Type:	Package
Title:	Sparsed Sliced Inverse Regression via Lasso
Version:	1.0
Date:	2025-03-31
Authors@R:	c(person(given = "Zhigen", family = "Zhao", role = c("aut", "cre"), email = "zhigen.zhao@gmail.com"), perso

Maintainer:	Zhigen Zhao <zhigen.zhao@gmail.com>
Description:	Estimate the sufficient dimension reduction space using sparsed sliced inverse regression via Lasso (Lasso-SIR
License:	GPL-3
Imports:	glmnet, graphics, stats
Author:	Zhigen Zhao [aut, cre], Qian Lin [aut], Jun Liu [aut]

Index of help topics:

```
LassoSIR                LassoSIR
LassoSIR-Package        Sparsed Sliced Inverse Regression via Lasso
predict_Lasso_SIR       Prediction based on Lasso SIR
```

LassoSIR

Author(s)

Zhigen Zhao [aut, cre], Qian Lin [aut], Jun Liu [aut]

Maintainer: Zhigen Zhao <zhigen.zhao@gmail.com>

References

Lin, Q., Zhao, Z. , and Liu, J. (2018) On consistency and sparsity for sliced inverse regression in high dimension. Annals of Statistics. Vol. 46, Number 2. Page 580-610.

Lin, Q., Zhao, Z. , and Liu, J. (2019) Sparse Sliced Inverse Regression for High Dimensional Data. Journal of the American Statistical Association. Vol. 114, Number 528, Page 1726-1739.

See Also

NA

Examples

```
p <- 10
n <- 200


H <- 20
m <- n/H

beta <- array(0, c(p, 1) )
beta[1:3,1] <- rnorm(3, 0, 1)

X <- array(0, c(n, p ) )

rho <- 0.3
Sigma <- diag(p)
elements <- rho^(c((p-1):0,1:(p-1) ) )
for(i in 1:p)
    Sigma[i,] <- elements[(p+1-i):(2*p-i) ]
```

```
X <- matrix( rnorm(p*n), c(n, p) )
X <- X%*% chol(Sigma)

Y <-  ( X%*% beta )^3/2 + rnorm(n,0,1)
sir.lasso <- LassoSIR( X, Y, H, choosing.d="automatic",
          solution.path=FALSE, categorical=FALSE, nfolds=10,
          screening=FALSE)
beta.hat <- sir.lasso$beta/sqrt( sum( sir.lasso$beta^2 ) )
```

`predict_Lasso_SIR` *Prediction based on Lasso SIR*

Description

This function calculates the value of $X\beta$ for a new data set.

Usage

```
predict_Lasso_SIR( lassosirobj, newdata )
```

Arguments

`lassosirobj`	LassoSIR object when running the function LassoSIR.
`newdata`	A data frame consisting of the values of the predictors.

Details

Based on the estiamted β, this function provides the value $X\beta$ for any new input.

Value

predict_value = predict_value, beta = lassosirobj$beta, no.dim = lassosirobj$no.dim

`predict_value`	the value of $X\beta$.
`beta`	the estiamted value of the β.
`no.dim`	the dimension of the central space.

Author(s)

Zhigen Zhao, Qian Lin, Jun S. Liu

References

Lin, Q., Zhao, Z. , and Liu, J. (2017) On consistency and sparsity for sliced inverse regression in high dimension. Annals of Statistics.

Lin, Q., Zhao, Z. , and Liu, J. (2016) Sparse Sliced Inverse Regression for High Dimensional Data.

Examples

```
p <- 10
n <- 200


H <- 20
m <- n/H

beta <- array(0, c(p, 1) )
beta[1:3,1] <- rnorm(3, 0, 1)

X <- array(0, c(n, p ) )

rho <- 0.3
Sigma <- diag(p)
elements <- rho^(c((p-1):0,1:(p-1) ) )
for(i in 1:p)
    Sigma[i,] <- elements[(p+1-i):(2*p-i) ]


X <- matrix( rnorm(p*n), c(n, p) )
X <- X%*% chol(Sigma)

Y <-  ( X%*% beta )^3/2 + rnorm(n,0,1)
sir.lasso <- LassoSIR( X, Y, H, choosing.d="automatic",
          solution.path=FALSE, categorical=FALSE, nfolds=10,
          screening=FALSE)

res = predict_Lasso_SIR( sir.lasso, newdata=data.frame( matrix( rnorm(5*p), c(5, p) )  ) )
```

Index