Package 'SILFS'

January 20, 2025

Type Package

Title Subgroup Identification with Latent Factor Structure

Version 0.1.0

Author Yong He [aut],
Dong Liu [aut],
Fuxin Wang [aut, cre],
Mingjuan Zhang [aut],
Wenxin Zhou [aut]

Maintainer Fuxin Wang <wangfuxin2001@163.com>

Description In various domains, many datasets exhibit both high variable dependency and group structures, which necessitates their simultaneous estimation. This package provides functions for two subgroup identification methods based on penalized functions, both of which utilize factor model structures to adapt to data with cross-sectional dependency. The first method is the Subgroup Identification with Latent Factor Structure Method (SILFSM) we proposed. By employing Center-Augmented Regularization and factor structures, the SILFSM effectively eliminates data dependencies while identifying subgroups within datasets. For this model, we offer optimization functions based on two different methods: Coordinate Descent and our newly developed Difference of Convex-Alternating Direction Method of Multipliers (DC-ADMM) algorithms; the latter can be applied to cases where the distance function in Center-Augmented Regularization takes L1 and L2 forms. The other method is the Factor-Adjusted Pairwise Fusion Penalty (FA-PFP) model, which incorporates factor augmentation into the Pairwise Fusion Penalty (PFP) developed by Ma, S. and Huang, J. (2017) <doi:10.1080/01621459.2016.1148039>. Additionally, we provide a function for the Standard CAR (S-CAR) method, which does not consider the dependency and is for comparative analysis with other approaches. Furthermore, functions based on the Bayesian Information Criterion (BIC) of the SILFSM and the FA-PFP method are also included in 'SILFS' for selecting tuning parameters. For more details of Subgroup Identification with Latent Factor Structure Method, please refer to He et al. (2024) <doi:10.48550/arXiv.2407.00882>.

License GPL-2 | GPL-3

Encoding UTF-8

RoxygenNote 7.3.1

Imports MASS, glmnet, stats, Ckmeans.1d.dp

NeedsCompilation no

Repository CRAN

Date/Publication 2024-07-03 16:20:06 UTC

Contents

BIC_PFP 2
BIC_SILFS 4
DCADMM_iter_l1 5
DCADMM_iter_l2 7
FA_PFP 9
INIT 10
SCAR 11
SILFS 12

Index **14**

BIC_PFP *Selecting Tuning Parameter for Factor Adjusted-Pairwise Fusion Penalty (FA-PFP) Method via corresponding BIC*

Description

This function is to select tuning parameters simultaneously for FA-PFP method via minimizing the BIC.

Usage

```
BIC_PFP(
  Y,
  Fhat,
  Uhat,
  alpha_init,
  lasso_start,
  lasso_stop,
  lam_start,
  lam_stop,
  grid_1,
  grid_2,
  epsilon
)
```

Arguments

`Y` The response vector of length n.

`Fhat` The estimated common factors matrix of size $n \times r$.

`Uhat` The estimated idiosyncratic factors matrix of size $n \times p$.

`alpha_init` The initialization of intercept parameter.

`lasso_start` The user-supplied start search value of the tuning parameters for LASSO.

`lasso_stop` The user-supplied stop search value of the tuning parameters for LASSO.

`lam_start` The user-supplied start search value of the tuning parameters for Pairwise Fusion Penalty.

`lam_stop` The user-supplied stop search value of the tuning parameters for Pairwise Fusion Penalty.

`grid_1` The user-supplied number of search grid points corresponding to the LASSO tuning parameter.

`grid_2` The user-supplied number of search grid points corresponding to the tuning parameter for Pairwise Fusion Penalty.

`epsilon` The user-supplied stopping tolerance.

Value

A list with the following components:

`lasso` The tuning parameter of the LASSO penalty selected using BIC.

`lambda` The tuning parameter of the Pairwise Concave Fusion Penalty selected using BIC.

Author(s)

Yong He, Liu Dong, Fuxin Wang, Mingjuan Zhang, Wenxin Zhou.

Examples

```
n <- 50
p <- 50
r <- 3
lasso_start <- sqrt(log(p)/n)*0.1
lasso_stop <- sqrt(log(p)/n)
lam_start <- 0.3
lam_stop <- 1
grid_1 <- 5
grid_2 <- 5
alpha <- sample(c(-3,3),n,replace=TRUE,prob=c(1/2,1/2))
beta <- c(rep(1,2),rep(0,48))
B <- matrix((rnorm(p*r,1,1)),p,r)
F_1 <- matrix((rnorm(n*r,0,1)),n,r)
U <- matrix(rnorm(p*n,0,0.1),n,p)
X <- F_1%*%t(B)+U
Y <- alpha + X%*%beta + rnorm(n,0,0.5)
```

```
alpha_init <- INIT(Y,F_1,0.1)

BIC_PFP(Y,F_1,U,alpha_init,lasso_start,lasso_stop,lam_start,lam_stop,grid_1,grid_2,0.3)
```

BIC_SILFS *Selecting Tuning Parameter for SILFS Method via corresponding BIC*

Description

This function is to select tuning parameters simultaneously for SILFS method via minimizing the BIC.

Usage

```
BIC_SILFS(
  Y,
  Fhat,
  Uhat,
  K,
  alpha_init,
  lasso_start,
  lasso_stop,
  CAR_start,
  CAR_stop,
  grid_1,
  grid_2,
  epsilon
)
```

Arguments

Y	The response vector of length n.
Fhat	The estimated common factors matrix of size $n \times r$.
Uhat	The estimated idiosyncratic factors matrix of size $n \times p$.
K	The estimated subgroup number.
alpha_init	The initialization of intercept parameter.
lasso_start	The user-supplied start search value of the tuning parameters for LASSO.
lasso_stop	The user-supplied stop search value of the tuning parameters for LASSO.
CAR_start	The user-supplied start search value of the tuning parameters for Center-Augmented Regularization.
CAR_stop	The user-supplied stop search value of the tuning parameters for Center-Augmented Regularization.
grid_1	The user-supplied number of search grid points corresponding to the LASSO tuning parameter.

`grid_2` The user-supplied number of search grid points corresponding to the tuning parameter for Center-Augmented Regularization.

`epsilon` The user-supplied stopping tolerance.

Value

A list with the following components:

`lasso` The tuning parameter of the LASSO penalty selected using BIC.

`CAR` The tuning parameter of the Center Augmented Regularization selected using BIC.

Examples

```
n <- 50
p <- 50
r <- 3
K <- 2
lasso_start <- sqrt(log(p)/n)*0.01
lasso_stop <- sqrt(log(p)/n)*10^(0.5)
CAR_start <- 0.001
CAR_stop <- 0.1
grid_1 <- 5
grid_2 <- 5
alpha <- sample(c(-3,3),n,replace=TRUE,prob=c(1/2,1/2))
beta <- c(rep(1,2),rep(0,48))
B <- matrix((rnorm(p*r,1,1)),p,r)
F_1 <- matrix((rnorm(n*r,0,1)),n,r)
U <- matrix(rnorm(p*n,0,0.1),n,p)
X <- F_1%*%t(B)+U
Y <- alpha + X%*%beta + rnorm(n,0,0.5)
alpha_init <- INIT(Y,F_1,0.1)

BIC_SILFS(Y,F_1,U,K,alpha_init,lasso_start,lasso_stop,CAR_start,CAR_stop,grid_1,grid_2,0.3)
```

`DCADMM_iter_l1` *SILFS-Based Subgroup Identification and Variable Selection Optimized by DC-ADMM under the L1 Distance*

Description

This function employs SILFS method and uses the corresponding Difference of Convex functions-Alternating Direction Method of Multipliers (DC-ADMM) algorithm for optimization to identify subgroup structures and conduct variable selection under the L1 Distance.

Usage

```
DCADMM_iter_l1(
  Y,
  F_hat,
  U_hat,
  r_1,
  r_2,
  r_3,
  lambda_1,
  lambda_2,
  K,
  alpha_init,
  epsilon_1,
  epsilon_2
)
```

Arguments

`Y`	The response vector of length n.
`F_hat`	The estimated factor matrix of size $n \times r$.
`U_hat`	The estimated idiosyncratic factors matrix of size $n \times p$.
`r_1`	The Lagrangian augmentation parameter for constraints of intercepts.
`r_2`	The Lagrangian augmentation parameter for constraints of group centers.
`r_3`	The Lagrangian augmentation parameter for constraints of coefficients.
`lambda_1`	The tuning parameter for Center-Augmented Regularization.
`lambda_2`	The tuning parameter for LASSO.
`K`	The estimated group number.
`alpha_init`	The initialization of intercept parameter.
`epsilon_1`	The user-supplied stopping error for outer loop.
`epsilon_2`	The user-supplied stopping error for inner loop.

Value

A list with the following components:

`alpha_curr`	The estimated intercept parameter vector of length n.
`gamma_curr`	The estimated vector of subgroup centers of length K.
`theta_curr`	The estimated regression coefficient vector, matched with common factor terms, with a dimension of r.
`beta_curr`	The estimated regression coefficients matched with idiosyncratic factors, with a dimension of p.

Author(s)

Yong He, Liu Dong, Fuxin Wang, Mingjuan Zhang, Wenxin Zhou.

References

He, Y., Liu, D., Wang, F., Zhang, M., Zhou, W., 2024. High-Dimensional Subgroup Identification under Latent Factor Structures.

Examples

```
n <- 50
p <- 50
r <- 3
K <- 2
alpha <- sample(c(-3,3),n,replace=TRUE,prob=c(1/2,1/2))
beta <- c(rep(1,2),rep(0,48))
B <- matrix((rnorm(p*r,1,1)),p,r)
F_1 <- matrix((rnorm(n*r,0,1)),n,r)
U <- matrix(rnorm(p*n,0,0.1),n,p)
X <- F_1%*%t(B)+U
Y <- alpha + X%*%beta + rnorm(n,0,0.5)
alpha_init <- INIT(Y,F_1,0.1)
DCADMM_iter_l1(Y,F_1,U,0.5,0.5,0.5,0.01,0.05,K,alpha_init,1,0.3)
```

DCADMM_iter_l2 *SILFS-Based Subgroup Identification and Variable Selection Optimized by DC-ADMM under the L2 Distance*

Description

This function employs SILFS method and uses the corresponding Difference of Convex functions-Alternating Direction Method of Multipliers (DC-ADMM) algorithm for optimization to identify subgroup structures and conduct variable selection under the L2 Distance.

Usage

```
DCADMM_iter_l2(
  Y,
  F_hat,
  U_hat,
  r_1,
  r_2,
  r_3,
  lambda_1,
  lambda_2,
  K,
  alpha_init,
  epsilon_1,
  epsilon_2
)
```

Arguments

Y The response vector of length n.

F_hat The estimated factor matrix of size $n \times r$.

U_hat The estimated idiosyncratic factors matrix of size $n \times p$.

r_1 The Lagrangian augmentation parameter for constraints of intercepts.

r_2 The Lagrangian augmentation parameter for constraints of group centers.

r_3 The Lagrangian augmentation parameter for constraints of coefficients.

lambda_1 The tuning parameter for Center-Augmented Regularization.

lambda_2 The tuning parameter for LASSO.

K The estimated group number.

alpha_init The initialization of intercept parameter.

epsilon_1 The user-supplied stopping error for outer loop.

epsilon_2 The user-supplied stopping error for inner loop.

Value

A list with the following components:

alpha_curr The estimated intercept parameter vector of length n.

gamma_curr The estimated vector of subgroup centers of length K.

theta_curr The estimated regression coefficient vector, matched with common factor terms, with a dimension of r.

beta_curr The estimated regression coefficients matched with idiosyncratic factors, with a dimension of p.

Author(s)

Yong He, Liu Dong, Fuxin Wang, Mingjuan Zhang, Wenxin Zhou.

References

He, Y., Liu, D., Wang, F., Zhang, M., Zhou, W., 2024. High-Dimensional Subgroup Identification under Latent Factor Structures.

Examples

```
n <- 50
p <- 50
r <- 3
K <- 2
alpha <- sample(c(-3,3),n,replace=TRUE,prob=c(1/2,1/2))
beta <- c(rep(1,2),rep(0,48))
B <- matrix((rnorm(p*r,1,1)),p,r)
F_1 <- matrix((rnorm(n*r,0,1)),n,r)
U <- matrix(rnorm(p*n,0,0.1),n,p)
X <- F_1%*%t(B)+U
```

```
Y <- alpha + X%*%beta + rnorm(n,0,0.5)
alpha_init <- INIT(Y,F_1,0.1)
DCADMM_iter_l2(Y,F_1,U,0.5,0.5,0.5,0.01,0.05,K,alpha_init,1,0.3)
```

FA_PFP *Factor Adjusted-Pairwise Fusion Penalty (FA-PFP) Method for Subgroup Identification and Variable Selection*

Description

This function utilizes the FA-PFP method implemented via the Alternating Direction Method of Multipliers (ADMM) algorithm to identify subgroup structures and conduct variable selection.

Usage

```
FA_PFP(Y, Fhat, Uhat, vartheta, lam, gam, alpha_init, lam_lasso, epsilon)
```

Arguments

`Y` The response vector of length n.

`Fhat` The estimated common factors matrix of size $n \times r$.

`Uhat` The estimated idiosyncratic factors matrix of size $n \times p$.

`vartheta` The Lagrangian augmentation parameter for intercepts.

`lam` The tuning parameter for Pairwise Fusion Penalty.

`gam` The user-supplied parameter for Alternating Direction Method of Multipliers (ADMM) algorithm.

`alpha_init` The initialization of intercept parameter.

`lam_lasso` The tuning parameter for LASSO.

`epsilon` The user-supplied stopping tolerance.

Value

A list with the following components:

`alpha_m` The estimated intercept parameter vector of length n.

`theta_m` The estimated regression coefficient vector, matched with common factor terms, with a dimension of r.

`beta_m` The estimated regression coefficients matched with idiosyncratic factors, with a dimension of p.

`eta_m` A numeric matrix storing the pairwise differences of the estimated intercepts, with size of $n \times (n \times (n-1)/2)$.

Author(s)

Yong He, Liu Dong, Fuxin Wang, Mingjuan Zhang, Wenxin Zhou.

References

Ma, S., Huang, J., 2017. A concave pairwise fusion approach to subgroup analysis.

Examples

```
n <- 50
p <- 50
r <- 3
alpha <- sample(c(-3,3),n,replace=TRUE,prob=c(1/2,1/2))
beta <- c(rep(1,2),rep(0,48))
B <- matrix((rnorm(p*r,1,1)),p,r)
F_1 <- matrix((rnorm(n*r,0,1)),n,r)
U <- matrix(rnorm(p*n,0,0.1),n,p)
X <- F_1%*%t(B)+U
Y <- alpha + X%*%beta + rnorm(n,0,0.5)
alpha_init <- INIT(Y,F_1,0.1)
FA_PFP(Y,F_1,U,1,0.67,3,alpha_init,0.05,0.3)
```

INIT *Initialization Function for the Intercept Parameter*

Description

This function computes initial values for intercept parameter by solving a ridge regression problem.

Usage

```
INIT(Y, X, lam_ridge)
```

Arguments

Y	The response vector of length n.
X	The design matrix of size $n \times p$.
lam_ridge	The tuning parameter for ridge regression.

Value

A numeric vector of length n, representing the initial estimation for intercept parameter.

Examples

```
n <- 100
p <- 100
beta <- rep(1,p)
X <- matrix(rnorm(100*100), n, p)
Y <- sample(c(-3,3),n,replace=TRUE,prob=c(1/2,1/2)) +  X%*%beta
lam_ridge <- 0.1
alpha_init <- INIT(Y, X, lam_ridge)
```

SCAR *Standard Center Augmented Regularization (S-CAR) Method for Subgroup Identification and Variable Selection*

Description

This function employs the S-CAR method under L2 distance and uses the Coordinate Descent Algorithm for optimization to identify subgroup structures and execute variable selection.

Usage

```
SCAR(Y, X, lam_CAR, lam_lasso, alpha_init, K, epsilon)
```

Arguments

`Y`	The response vector of length n.
`X`	The design matrix of size $n \times p$.
`lam_CAR`	The tuning parameter for Center-Augmented Regularization.
`lam_lasso`	The tuning parameter for lasso.
`alpha_init`	The initialization of intercept parameter.
`K`	The estimated group number.
`epsilon`	The user-supplied stopping tolerance.

Value

A list with the following components:

`alpha_m`	The estimated intercept parameter vector of length n.
`gamma`	The estimated vector of subgroup centers of length K.
`beta_m`	The estimated regression coefficient vector of dimension p.

Author(s)

Yong He, Liu Dong, Fuxin Wang, Mingjuan Zhang, Wenxin Zhou.

Examples

```
n <- 50
p <- 50
r <- 3
K <- 2
alpha <- sample(c(-3,3),n,replace=TRUE,prob=c(1/2,1/2))
beta <- c(rep(1,2),rep(0,48))
B <- matrix((rnorm(p*r,1,1)),p,r)
F_1 <- matrix((rnorm(n*r,0,1)),n,r)
U <- matrix(rnorm(p*n,0,0.1),n,p)
```

```
X <- F_1%*%t(B)+U
Y <- alpha + X%*%beta + rnorm(n,0,0.5)
alpha_init <- INIT(Y,X,0.1)
SCAR(Y,X,0.01,0.05,alpha_init,K,0.3)
```

SILFS *SILFS-Based Subgroup Identification and Variable Selection Optimized by Coordinate Descent under the L2 Distance*

Description

This function employs SILFS method under L2 distance and uses the Coordinate Descent Algorithm for optimization to effectively identify subgroup structures and perform variable selection.

Usage

```
SILFS(Y, X_aug, r, lam_CAR, lam_lasso, alpha_init, K, epsilon)
```

Arguments

`Y` The response vector of length n.

`X_aug` The augmented design matrix created by row concatenation of common and idiosyncratic factor matrices, with a size of $n \times (r + p)$.

`r` The user supplied number of common factors.

`lam_CAR` The tuning parameter for Center-Augmented Regularization.

`lam_lasso` The tuning parameter for LASSO.

`alpha_init` The initialization of intercept parameter.

`K` The user-supplied group number.

`epsilon` The user-supplied stopping tolerance.

Value

A vector containing the following components:

`alpha_m` The estimated intercept parameter vector of length n.

`gamma` The estimated vector of subgroup centers of length K.

`theta_m` The estimated regression coefficient vector, matched with common factor terms, with a dimension of r.

`beta_m` The estimated regression coefficients matched with idiosyncratic factors, with a dimension of p.

Author(s)

Yong He, Liu Dong, Fuxin Wang, Mingjuan Zhang, Wenxin Zhou.

References

He, Y., Liu, D., Wang, F., Zhang, M., Zhou, W., 2024. High-Dimensional Subgroup Identification under Latent Factor Structures.

Examples

```
n <- 50
p <- 50
r <- 3
K <- 2
alpha <- sample(c(-3,3),n,replace=TRUE,prob=c(1/2,1/2))
beta <- c(rep(1,2),rep(0,48))
B <- matrix((rnorm(p*r,1,1)),p,r)
F_1 <- matrix((rnorm(n*r,0,1)),n,r)
U <- matrix(rnorm(p*n,0,0.1),n,p)
X <- F_1%*%t(B)+U
Y <- alpha + X%*%beta + rnorm(n,0,0.5)
alpha_init <- INIT(Y,F_1,0.1)
SILFS(Y,cbind(F_1,U),3,0.01,0.05,alpha_init,K,0.3)
```

Index

BIC_PFP, 2
BIC_SILFS, 4

DCADMM_iter_l1, 5
DCADMM_iter_l2, 7

FA_PFP, 9

INIT, 10

SCAR, 11
SILFS, 12