Package 'Mulste'

August 9, 2022

Type Package

Title Mulste

Version 0.1.0

Author Xu Liu [aut,cre],
Yiming Liu [aut],
Xiao Zhang [aut],
Xingjie Shi [ctb]

Maintainer Xu Liu <liu.xu@sufe.edu.cn>

Description High-dimensional multi-view learning via structural tensor estimation. The B-splines are used to approximate component functions. The spline coefficients are rearranged into multiple third-order tensors or even a fourth-order tensor with low-rankness under some specific conditions. Dimension reduction can be achieved by Tucker decomposition and group sparse penalty, for example, LASSO, MCP or SCAD.

License GPL (>= 2)

Imports splines, Rcpp (>= 0.11.15), RcppEigen (>= 0.3.2.3.0)

LinkingTo Rcpp, RcppEigen

RoxygenNote 6.0.1

NeedsCompilation yes

Repository github

URL https://github.com/xliusufe/Mulste

Encoding UTF-8

R topics documented:

`Mulste-package` *High-dimensional multi-view learning via structural tensor estimation*

Description

For an integrative multi-view multivariate additive model (MARM), the B-splines are applied to approximate the component functions. We treat the coefficients as multiple third-order tensors or even a fourth-order tensor in some specific conditions (called structural MARM). The composed model can also be used when the number of covariates in each view is not equal. With the tensor low-rankness, the Tucker decomposition and group sparse penalty (lasso, mcp or scad) reduce the number of parameters. An alternative updating algorithm based on the coordinate descent strategy is used to estimate the core tensors and factor matrices, and further additive functions.

Details

This pacakge includes six main functions and six generating functions. `marm3` and `marm3.dr` yield the estimator of MARM. The difference is the former requires the fixed ranks and the latter can search the optimal ranks and regularization parameter simultaneously by BIC or CV method. `marm3.sim.fbs` and `marm3.sim.fsin` generate data of scenario I and II respectively. Scenario I assumes that the true functions are exactly residing in the space of B-spline basis functions. Scenario II assumes that the true functions are some linear combination of $sin(2\pi x)$ and $cos(\pi x)$. Similarly, `marm4` and `marm4.dr` yield the estimator of structural MARM. `marm4.sim.fbs` and `marm4.sim.fsin` are two generating functions of structural MARM with scenario I and II settings. `marmComposed` and `marmComposed.dr` yield the estimator of composed model. `marmComposed.sim.fbs` and `marmComposed.sim.fsin` are two generating functions of composed model with scenario I and II settings. They all have the same assumptions as MARM.

Author(s)

Xu Liu

Maintainer: Xu Liu <liu.xu@sufe.edu.cn>

References

High-dimensional multi-view learning via structural tensor estimation.

`marm3` *Fit MARM with sparsity assumption and fixed ranks.*

Description

Fit a multivariate additive model for multi-view data (MARM) using B-splines with given ranks (r_{1g}, r_{2g}, r_{3g}). Multiple third-order coefficient tensors can be estimated by this function. The group sparse penalty such as LASSO, MCP or SCAD and the coordinate descent algorithm are used to yield a sparsity estimator. The BIC or cross-validation method are used to search the optimal regularization parameter.

Usage

```
marm3 <- function(Y,X,group=NULL,K=6,r1=NULL,r2=NULL,r3=NULL,
                  method="BIC",ncv=10,penalty="LASSO",lambda=NULL,D0=NULL,
                  intercept=TRUE,degr=3,nlam=20,lam_min=0.01,
                  eps=1e-4,max_step=20, eps1=1e-4,max_step1=20,
                  gamma=2,dfmax=NULL,alpha=1)
```

Arguments

Y	A $n \times q$ numeric matrix of responses.
X	A $n \times p$ numeric design matrix for the model, where $p = \sum_g p_g$.
group	A p vector of the grouping index of predictors, e.g., $group = c(1,1,1,2,2,2)$ means there are 6 perdictors in the model, and the first three perdictors are in the same group and the last three perdictors are in another one. By default, we set $group = rep(1,p)$.
K	The number of B-spline basis functions, that is the plus of both degrees of basis functioins and the number of knots. Default is 6, which means cubic spline.
r1	The first dimension of single value matrix of the tensor. Default is 2.
r2	The second dimension of single value matrix of the tensor. Default is 2.
r3	The third dimension of single value matrix of the tensor. Default is 2.
method	The method to be applied to select regularization parameters. Either BIC (default), or CV.
ncv	The number of cross-validation folds. Default is 10. If method is not CV, ncv is useless.
penalty	The penalty to be applied to the model. Either LASSO (the default), MCP or SCAD.
lambda	A user-specified sequence of lambda values. By default, a sequence of values of length nlam is computed, equally spaced on the log scale.
D0	A user-specified list of initialized values, including ng sub-lists where ng is the number of groups. For each sub-list, it has four initialized matrix S_{(3)} (called S), A, B and C. By default, a list of initialization satisfying fixed ranks is computed by random.
intercept	A logical value indicating whether the intercept is fitted. Default is TRUE or set to zero by FALSE.
degr	The number of knots of B-spline base function. Default is 3.
nlam	The number of lambda values. Default is 20.
lam_min	The smallest value for lambda, as a fraction of lambda.max. Default is 0.01.
eps	Convergence threshhold. The algorithm iterates until the relative change in any coefficient is less than eps. Default is 1e-4.
max_step	Maximum number of iterations. Default is 20.
eps1	Convergence threshhold. The Coordinate descent method algorithm iterates until the relative change in any coefficient is less than eps1. Default is 1e-4.
max_step1	The maximum iterates number of coordinate descent method. Default is 20.
gamma	The tuning parameter of the MCP/SCAD penalty.
dfmax	Upper bound for the number of nonzero coefficients. Default is no upper bound. However, for large data sets, computational burden may be heavy for models with a large number of nonzero coefficients.

alpha	Tuning parameter for the Mnet estimator which controls the relative contributions from the LASSO, MCP or SCAD penalty and the ridge, or L2 penalty. alpha=1 is equivalent to LASSO, MCP or SCAD penalty, while alpha=0 would be equivalent to ridge regression. However, alpha=0 is not supported; alpha may be arbitrarily small, but not exactly 0.

Details

This function gives pq functional coefficients' estimators of marm. Multiple third-order tensors with multiple ranks (r_{1g}, r_{2g}, r_{3g}) need to be estimated. We fix these ranks and use an alternative updating algorithm to update its core tensor and factor matrices based on Tucker decomposition. Group LASSO, SCAD or MCP penalty is applied on the row of each factor matrix A^g to achieve variable selection.

Value

D	Estimator of coefficients $D_{(3)} = (D^1_{(3)}, ..., D^{ng}_{(3)})$ where ng is the number of groups.
mu	Estimator of intercept μ.
S.opt	A length-ng list including estimator of the core tensor $S_{(3)}$ of each coefficient tensor.
A.opt	A length-ng list including estimator of the factor matrix A of each coefficient tensor.
B.opt	A length-ng list including estimator of the factor matrix B of each coefficient tensor.
C.opt	A length-ng list including estimator of the factor matrix C of each coefficient tensor.
lambda.seq	The sequence of regularization parameter values in the path.
lambda_opt	The value of lambda with the minimum BIC or CV value.
rss	Residual sum of squares (RSS).
df	Degrees of freedom.
activeX	The active set of X. A length-p vector.
opts	Other related parameters used in algorithm. Some of them are set by default.
opts_pen	Other related parameters used in algorithm (especially parameters in penalty). Some of them are set by default.

References

High-dimensional multi-view learning via structural tensor estimation.

See Also

marm3_dr

Examples

```
library(Mulste)
n <- 200; q <- 5; p <- 100; s <- 3; ng <- 4
group <- rep(1:ng,each=p/ng)
mydata <- marm3.sim.fbs(n,q,p,s,group,isfixedR=1)
fit <- with(mydata, marm3(Y,X,group,K,r10,r20,r30,D0=D0,nlam=5))
```

marm3.dr *Fit MARM with sparsity assumption and unknown ranks.*

Description

Fit a multivariate additive model for multi-view data (MARM) using B-splines with unknown ranks (r_{1g}, r_{2g}, r_{3g}). Multiple third-order coefficient tensors can be estimated by this function. The group sparse penalty such as LASSO, MCP or SCAD and the coordinate descent algorithm are used to yield a sparsity estimator. The BIC or cross-validation method are used to search the optimal regularization parameter, multiple ranks and the number of B-spline basis functions simultaneously.

Usage

```
marm3.dr <- function(Y,X,group=NULL,K_index=NULL,r1_index=NULL,r2_index=NULL,
                     r3_index=NULL,method="BIC",ncv=10,penalty="LASSO",
                     lambda=NULL,D0=NULL,intercept=TRUE,nlam=50,degr=3,
                     lam_min=0.01,eps=1e-4,max_step=20,eps1=1e-4,
                     max_step1=20,gamma=2,dfmax=NULL,alpha=1)
```

Arguments

Y	A $n \times q$ numeric matrix of responses.
X	A $n \times p$ numeric design matrix for the model, where $p = \sum_g p_g$
group	A p vector of the grouping index of predictors, e.g., $group = c(1,1,1,2,2,2)$ means there are 6 perdictors in the model, and the first three perdictors are in the same group and the last three perdictors are in another one. By default, we set $group = rep(1,p)$.
K_index	The user-specified sequence of K. Default is a length-1 vector 6.
r1_index	A user-specified sequence of r_1 values, where r_1 is the first dimension of the tensor. Default is r1_index$= 1, \cdots, \min(\lceil \log(n) \rceil, p)$.
r2_index	A user-specified sequence of r_2 values, where r_2 is the second dimension of the tensor. Default is r2_index$= 1, \cdots$,max{K_index}.
r3_index	A user-specified sequence of r_3 values, where r_3 is the third dimension of the tensor. Default is r3_index$= 1, \cdots, \min(\lceil \log(n) \rceil, q)$.
method	The method to be applied to select the number of B-spline basis functions, regularization parameters and multiple ranks simultaneously. Either BIC (default), or CV.
ncv	The number of cross-validation folds. Default is 10. If method is not CV, ncv is useless.
penalty	The penalty to be applied to the model. Either LASSO (the default), MCP or SCAD.
lambda	A user-specified sequence of lambda values. By default, a sequence of values of length nlam is computed, equally spaced on the log scale.
D0	A user-specified list of initialized values, including ng sub-lists where ng is the number of groups. For each sub-list, it has four initialized matrix $S_{(3)}$ (called S), A, B and C. By default, a list of initialization satisfying fixed ranks is computed by random.
intercept	A logical value indicating whether the intercept is fitted. Default is TRUE or set to zero by FALSE.

degr	The number of knots of B-spline base function. Default is 3.
nlam	The number of lambda values. Default is 50.
lam_min	The smallest value for lambda, as a fraction of lambda.max. Default is 0.01.
eps	Convergence threshhold. The algorithm iterates until the relative change in any coefficient is less than eps. Default is 1e-4.
max_step	Maximum number of iterations. Default is 20.
eps1	Convergence threshhold. The Coordinate descent method algorithm iterates until the relative change in any coefficient is less than eps1. Default is 1e-4.
max_step1	The maximum iterates number of coordinate descent method. Default is 20.
gamma	The tuning parameter of the MCP/SCAD penalty.
dfmax	Upper bound for the number of nonzero coefficients. Default is no upper bound. However, for large data sets, computational burden may be heavy for models with a large number of nonzero coefficients.
alpha	Tuning parameter for the Mnet estimator which controls the relative contributions from the LASSO, MCP or SCAD penalty and the ridge, or L2 penalty. alpha=1 is equivalent to LASSO, MCP or SCAD penalty, while alpha=0 would be equivalent to ridge regression. However, alpha=0 is not supported; alpha may be arbitrarily small, but not exactly 0.

Details

This function gives pq functional coefficients' estimators of marm. Multiple third-order tensors with unknown ranks (r_1, r_2, r_3) need to be estimated. The BIC or CV can be applied to select the optimal regularization parameters, multiple ranks and the number of B-spline basis functions simultaneously. An alternative updating algorithm can be used to update its core tensor and factor matrices based on Tucker decomposition. Group LASSO, SCAD or MCP penalty is applied on the row of each factor matrix A^g to achieve variable selection. Generally, the number of B-spline basis functions we need is fixed by 6, i.e., cubic splines are used to approximate the component functions.

Value

D	Estimator of coefficients $D_{(3)} = (D^1_{(3)}, ..., D^{ng}_{(3)})$.
mu	Estimator of intercept μ.
S.opt	A length-ng list including estimator of the core tensor $S_{(3)}$ of each coefficient tensor.
A.opt	A length-ng list including estimator of the factor matrix A of each coefficient tensor.
B.opt	A length-ng list including estimator of the factor matrix B of each coefficient tensor.
C.opt	A length-ng list including estimator of the factor matrix C of each coefficient tensor.
rk_opt	The optimal ranks and the number of B-spline basis functions that slected by BIC, or CV. It is a vector with length 4, which are selected r_1, r_2, r_3, and K.
lambda.seq	The sequence of regularization parameter values in the path.
lambda_opt	The value of lambda with the minimum BIC or CV value.
rss	Residual sum of squares (RSS).
df	Degrees of freedom.

activeX	The active set of X. A length-p vector.
opts	Other related parameters used in algorithm. Some of them are set by default.
opts_pen	Other related parameters used in algorithm (especially parameters in peanlty). Some of them are set by default.

References

High-dimensional multi-view learning via structural tensor estimation.

See Also

marm3

Examples

```
library(Mulste)
n <- 200; q <- 5; p <- 100; s <- 3; ng = 4
group <- rep(1:ng,each=p/ng)
mydata <- marm3.sim.fbs(n,q,p,s,group)
fit <- with(mydata, marm3.dr(Y,X,group,K,r1_index,r2_index,r3_index,D0=D0,nlam=5))
```

---

marm3.sim.fbs *Generate scenario I data from MARM model.*

Description

Generate scenario I data for MARM model.

Usage

```
marm3.sim.fbs <- function(n,q,p,s,group=NULL,r10=2,r20=2,r30=2,isfixedR=0,
                          D3=NULL,K=6,degr=3,sigma2=NULL,seed_id=NULL,
                          r1_index=NULL,r2_index=NULL,r3_index=NULL,D0=NULL)
```

Arguments

n	Sample size.
q	The number of responses, $q \geq 1$.
p	The number of covariates, $p \geq 1$.
s	The true covariates of each view associated with responses, $s \geq 1$.
group	A length-p vector of the grouping index of predictors, e.g., $group = c(1,1,1,2,2,2)$ means there are 6 perdictors in the model, and the first three perdictors are in the same group and the last three perdictors are in another one. By default, we set $group = rep(1,p)$.
r10	The first dimension of the tensor. Default is 2.
r20	The second dimension of the tensor. Default is 2.
r30	The third dimension of the tensor. Default is 2.
isfixedR	A logical value indicating whether ranks are fixed.
D3	The mode of unfolding $D_{(3)}$. By default, D3 is generated by random.

K	The number of B-spline basis functions, that is the plus of both degrees of basis functioins and the number of knots. Default is 6, which means cubic splines.
degr	The number of knots of B-spline base function. Default is 3.
sigma2	err variance. Default is 0.1.
seed_id	A positive integer, the seed for generating the random numbers. Default is 1000.
r1_index	A user-specified sequence of r_1 values, where r_1 is the first dimension of single value matrix of the tensor. Default is r1_index= $1, \cdots, \min(\lceil \log(n) \rceil, p)$. if $isfixedR = 1$, r1_index is useless.
r2_index	A user-specified sequence of r_2 values, where r_2 is the second dimension of single value matrix of the tensor. Default is r2_index= $1, \cdots$,max{K_index}. if $isfixedR = 1$, r2_index is useless.
r3_index	A user-specified sequence of r_3 values, where r_3 is the third dimension of single value matrix of the tensor. Default is r3_index= $1, \cdots, \min(\lceil \log(n) \rceil, q)$. if $isfixedR = 1$, r3_index is useless.
D0	A user-specified list of initialized values, including ng sub-lists where ng is the number of groups. For each sub-list, it has four initialized matrices $S_{(3)}$ (called S), A, B and C. By default, a list of initialization satisfying fixed ranks is computed by random.

Details

This function can generate scenario I data of marm model. isfixedR is required to yield a different initialization D0. In scenario I, the true functions are exactly residing in the space of B-spline basis functions.

Value

Y	Response, a $n \times q$-matrix.
X	Design matrix, a $n \times p$-matrix.
f0	True functions, a $n \times p$-matrix.
group	The grouping index of predictors, a length-p vector.
D0	The initialized values.
...	Other options for algorithm.

References

High-dimensional multi-view learning via structural tensor estimation.

See Also

marm3.sim.fsin

Examples

```
library(Mulste)
n <- 200; q <- 5; p <- 100; s <- 3; ng = 4
group <- rep(1:ng,each=p/ng)
mydata <- marm3.sim.fbs(n,q,p,s,group)
```

marm3.sim.fsin *Generate scenario II data from MARM model.*

Description

Generate scenario II data for a multivariate additive model for multi-view data.

Usage

```
marm3.sim.fsin <- function(n,q,p,s,group=NULL,r10=2,r20=2,r30=2,isfixedR=0,
                           D2=NULL,K=6,degr=3,sigma2=NULL,seed_id=NULL,
                           r1_index=NULL,r2_index=NULL,r3_index=NULL,D0=NULL)
```

Arguments

n	Sample size.
q	The number of responses, $q \geq 1$.
p	The number of covariates, $p \geq 1$.
s	The true covariates of each view associated with responses, $s \geq 1$.
group	A length-p vector of the grouping index of predictors, e.g., $group = c(1,1,1,2,2,2)$ means there are 6 perdictors in the model, and the first three perdictors are in the same group and the last three perdictors are in another one. By default, we set $group = rep(1,p)$.
r10	The first dimension of the tensor. Default is 2.
r20	The second dimension of the tensor. Default is 2.
r30	The third dimension of the tensor. Default is 2.
isfixedR	A logical value indicating whether ranks are fixed.
D2	The mode of unfolding $D_{(2)}$. By default, D2 is generated by random.
K	The number of B-spline basis functions, that is the plus of both degrees of basis functioins and the number of knots. Default is 6, which means cubic spline.
degr	The number of knots of B-spline base function. Default is 3.
sigma2	err variance. Default is 0.1.
seed_id	A positive integer, the seed for generating the random numbers. Default is 1000.
r1_index	A user-specified sequence of r_1 values, where r_1 is the first dimension of single value matrix of the tensor. Default is r1_index$= 1, \cdots, \min(\lceil \log(n) \rceil, p)$. if $isfixedR = 1$, r1_index is useless.
r2_index	A user-specified sequence of r_2 values, where r_2 is the second dimension of single value matrix of the tensor. Default is r2_index$= 1, \cdots$,max{K_index}. if $isfixedR = 1$, r2_index is useless.
r3_index	A user-specified sequence of r_3 values, where r_3 is the third dimension of single value matrix of the tensor. Default is r3_index$= 1, \cdots, \min(\lceil \log(n) \rceil, q)$. if $isfixedR = 1$, r3_index is useless.
D0	A user-specified list of initialized values, including ng sub-lists where ng is the number of groups. For each sub-list, it has four initialized matrices $S_{(3)}$ (called S), A, B and C. By default, a list of initialization satisfying fixed ranks is computed by random.

Details

This function can generate scenario II data of marm model. `isfixedR` is required to yield a different initialization `D0`. In scenario II, the true functions are the linear combination of $sin(2\pi x)$ and $cos(\pi x)$.

Value

`Y`	Response, a $n \times q$-matrix.
`X`	Design matrix, a $n \times p$-matrix.
`f0`	True functions, a $n \times p$-matrix.
`group`	The grouping index of predictors, a length-p vector.
`D0`	The initialized values.
`...`	Other options for algorithm.

References

High-dimensional multi-view learning via structural tensor estimation.

See Also

marm3.sim.fbs

Examples

```
library(Mulste)
n <- 200; q <- 5; p <- 100; s <- 3; ng = 4
group <- rep(1:ng,each=p/ng)
mydata <- marm3.sim.fsin(n,q,p,s,group)
```

`marm4` *Fit structural MARM with sparsity assumption and fixed ranks.*

Description

Fit a structural intergative multi-view multivariate additive model (structural marm) using B-splines with given ranks (r_1, r_2, r_3, r_4). A fourth-order coefficient tensor can be estimated by this function. The group sparse penalty such as `LASSO`, `MCP` or `SCAD` and the coordinate descent algorithm are used to yield a sparsity estimator. The `BIC` or `cross-validation` method are used to search the optimal regularization parameter.

Usage

```
marm4 <- function(Y,X,group=NULL,K=6,r1=NULL,r2=NULL,r3=NULL,r4=NULL,
                  method="BIC",ncv=10,penalty="LASSO",lambda=NULL,D0=NULL,
                  intercept=TRUE,nlam=20,degr=3,lam_min=0.01,
                  eps=1e-4,max_step=10,eps1=1e-4,max_step1=10,
                  gamma=2,dfmax=NULL,alpha=1)
```

Arguments

Y	A $n \times q$ numeric matrix of responses.
X	A $n \times p$ numeric design matrix for the model, where $p = \sum_g p_g$
group	A p vector of the grouping index of predictors, e.g., $group = c(1, 1, 1, 2, 2, 2)$ means there are 6 perdictors in the model, and the first three perdictors are in the same group and the last three perdictors are in another one. By default, we set $group = rep(1, p)$.
K	The number of B-spline basis functions, that is the plus of both degrees of basis functioins and the number of knots. Default is 6, which means cubic splines.
r1	The first dimension of single value matrix of the tensor. Default is 2.
r2	The second dimension of single value matrix of the tensor. Default is 2.
r3	The third dimension of single value matrix of the tensor. Default is 2.
r4	The fourth dimension of single value matrix of the tensor. Default is 2.
method	The method to be applied to select regularization parameters. Either BIC (default), or CV.
ncv	The number of cross-validation folds. Default is 10. If method is not CV, ncv is useless.
penalty	The penalty to be applied to the model. Either LASSO (the default), MCP or SCAD.
lambda	A user-specified sequence of lambda values. By default, a sequence of values of length nlam is computed, equally spaced on the log scale.
D0	A user-specified list of initialized values, including five initialized matrix $S_{(4)}$ (called S), A, B, C and D. By default, a list of initialization satisfying fixed ranks is computed by random.
intercept	A logical value indicating whether the intercept is fitted. Default is TRUE or set to zero by FALSE.
degr	The number of knots of B-spline base function. Default is 3.
nlam	The number of lambda values. Default is 20.
lam_min	The smallest value for lambda, as a fraction of lambda.max. Default is 0.01.
eps	Convergence threshhold. The algorithm iterates until the relative change in any coefficient is less than eps. Default is 1e-4.
max_step	Maximum number of iterations. Default is 20.
eps1	Convergence threshhold. The Coordinate descent method algorithm iterates until the relative change in any coefficient is less than eps1. Default is 1e-4.
max_step1	The maximum iterates number of coordinate descent method. Default is 20.
gamma	The tuning parameter of the MCP/SCAD penalty.
dfmax	Upper bound for the number of nonzero coefficients. Default is no upper bound. However, for large data sets, computational burden may be heavy for models with a large number of nonzero coefficients.
alpha	Tuning parameter for the Mnet estimator which controls the relative contributions from the LASSO, MCP/SCAD penalty and the ridge, or L2 penalty. alpha=1 is equivalent to LASSO, MCP/SCAD penalty, while alpha=0 would be equivalent to ridge regression. However, alpha=0 is not supported; alpha may be arbitrarily small, but not exactly 0.

Details

This function gives pq functional coefficients' estimators of marm. A fourth-order tensor with multiple ranks (r_1, r_2, r_3, r_4) need to be estimated. We fix these ranks and use an alternative updating algorithm to update its core tensor and factor matrices based on Tucker decomposition. Group LASSO, SCAD or MCP penalty is applied on the row of the factor matrix A to achieve variable selection.

Value

`D`	Estimator of $D_{(4)}$.
`mu`	Estimator of intercept μ.
`S.opt`	A length-ng list including estimator of the core tensor $S_{(3)}$ of each coefficient tensor.
`A.opt`	A length-ng list including estimator of the factor matrix A of each coefficient tensor.
`B.opt`	A length-ng list including estimator of the factor matrix B of each coefficient tensor.
`C.opt`	A length-ng list including estimator of the factor matrix C of each coefficient tensor.
`D.opt`	A length-ng list including estimator of the factor matrix D of each coefficient tensor.
`lambda.seq`	The sequence of regularization parameter values in the path.
`lambda_opt`	The value of `lambda` with the minimum `BIC` or `CV` value.
`rss`	Residual sum of squares (RSS).
`df`	Degrees of freedom.
`activeX`	The active set of X. A length-p vector.
`opts`	Other related parameters used in algorithm. Some of them are set by default.
`opts_pen`	Other related parameters used in algorithm (especially parameters in penalty). Some of them are set by default.

References

High-dimensional multi-view learning via structural tensor estimation.

See Also

marm4_dr

Examples

```
library(Mulste)
n <- 200; q <- 5; p <- 100; s <- 3; ng <- 4
group <- rep(1:ng,each=p/ng)
mydata <- marm4.sim.fbs(n,q,p,s,group,isfixedR=1)
fit <- with(mydata, marm4(Y,X,group,K,r10,r20,r30,r40,D0=D0,nlam=5))
```

marm4.dr *Fit structural MARM with sparsity assumption and unknown ranks.*

Description

Fit a structural multivariate additive model for multi-view data (structural MARM) using B-splines with unknown ranks (r_1, r_2, r_3, r_4). A fourth-order coefficient tensor can be estimated by this function. The group sparse penalty such as LASSO, MCP or SCAD and the coordinate descent algorithm are used to yield a sparsity estimator. The BIC or cross-validation method is used to search the optimal regularization parameter, multiple ranks and the number of B-spline basis functions simultaneously.

Usage

```
marm4.dr <- function(Y,X,group,K_index=NULL,r1_index=NULL,r2_index=NULL,
                     r3_index=NULL,r4_index=NULL,method="BIC",ncv=10,
                     penalty="LASSO",lambda=NULL,D0=NULL,
                     intercept=TRUE,nlam=20,degr=3,lam_min=0.01,
                     eps=1e-4,max_step=10,eps1=1e-4,max_step1=10,
                     gamma=2,dfmax=NULL,alpha=1)
```

Arguments

Y	A $n \times q$ numeric matrix of responses.
X	A $n \times p$ numeric design matrix for the model, where $p = \sum_g p_g$
group	A p vector of the grouping index of predictors, e.g., $group = c(1,1,1,2,2,2)$ means there are 6 perdictors in the model, and the first three perdictors are in the same group and the last three perdictors are in another one. By default, we set $group = rep(1, p)$.
K_index	The user-specified sequence of K. Default is a length-1 vector 6.
r1_index	A user-specified sequence of r_1 values, where r_1 is the first dimension of single value matrix of the tensor. Default is r1_index$= 1, \cdots, \min(\lceil \log(n) \rceil, p)$.
r2_index	A user-specified sequence of r_2 values, where r_2 is the second dimension of single value matrix of the tensor. Default is r2_index$= 1, \cdots$,max{K_index}.
r3_index	A user-specified sequence of r_3 values, where r_3 is the third dimension of single value matrix of the tensor. Default is r3_index$= 1, \cdots, \min(\lceil \log(n) \rceil, ng)$.
r4_index	A user-specified sequence of r_4 values, where r_4 is the third dimension of single value matrix of the tensor. Default is r4_index$= 1, \cdots, \min(\lceil \log(n) \rceil, q)$.
method	The method to be applied to search the number of B-spline basis functions, regularization parameters and multiple ranks simultaneously. Either BIC (default), or CV.
ncv	The number of cross-validation folds. Default is 10. If method is not CV, ncv is useless.
penalty	The penalty to be applied to the model. Either LASSO (the default), MCP or SCAD.
lambda	A user-specified sequence of lambda values. By default, a sequence of values of length nlam is computed, equally spaced on the log scale.

D0	A user-specified list of initialized values, including five initialized matrix $S_{(4)}$ (called S), A, B, C and D. By default, a list of initialization satisfying fixed ranks is computed by random.
intercept	A logical value indicating whether the intercept is fitted. Default is TRUE or set to zero by FALSE.
degr	The number of knots of B-spline base function. Default is 3.
nlam	The number of lambda values. Default is 20.
lam_min	The smallest value for lambda, as a fraction of lambda.max. Default is 0.01.
eps	Convergence threshhold. The algorithm iterates until the relative change in any coefficient is less than eps. Default is 1e-4.
max_step	Maximum number of iterations. Default is 20.
eps1	Convergence threshhold. The Coordinate descent method algorithm iterates until the relative change in any coefficient is less than eps1. Default is 1e-4.
max_step1	The maximum iterates number of coordinate descent method. Default is 20.
gamma	The tuning parameter of the MCP or SCAD penalty (see details).
dfmax	Upper bound for the number of nonzero coefficients. Default is no upper bound. However, for large data sets, computational burden may be heavy for models with a large number of nonzero coefficients.
alpha	Tuning parameter for the Mnet estimator which controls the relative contributions from the LASSO, MCP or SCAD penalty and the ridge, or L2 penalty. alpha=1 is equivalent to LASSO, MCP or SCAD penalty, while alpha=0 would be equivalent to ridge regression. However, alpha=0 is not supported; alpha may be arbitrarily small, but not exactly 0.

Details

This function gives pq functional coefficients' estimators of structural marm. A fourth-order tensor with unknown ranks (r_1, r_2, r_3, r_4) need to be estimated. The BIC or CV can be applied to select the optimal regularization parameter, multiple ranks and the number of B-spline basis functions simultaneously. An alternative updating algorithm can be used to update its core tensor and factor matrices based on Tucker decomposition. Group LASSO, SCAD or MCP penalty is applied on the row of the factor matrix A to achieve variable selection.

Value

D	Estimator of $D_{(4)}$.
mu	Estimator of intercept μ.
S.opt	A length-ng list including estimator of the core tensor $S_{(3)}$ of each coefficient tensor.
A.opt	A length-ng list including estimator of the factor matrix A of each coefficient tensor.
B.opt	A length-ng list including estimator of the factor matrix B of each coefficient tensor.
C.opt	A length-ng list including estimator of the factor matrix C of each coefficient tensor.
D.opt	A length-ng list including estimator of the factor matrix D of each coefficient tensor.

rk_opt	The optimal ranks and the number of B-spline basis functions that selected by BIC, or CV. It is a vector with length 5, which are selected r_1, r_2, r_3, r_4 and K.
lambda.seq	The sequence of regularization parameter values in the path.
lambda_opt	The value of lambda with the minimum BIC or CV value.
rss	Residual sum of squares (RSS).
df	Degrees of freedom.
activeX	The active set of X. A length-p vector.
opts	Other related parameters used in algorithm. Some of them are set by default.
opts_pen	Other related parameters used in algorithm (especially parameters in peanlty). Some of them are set by default.

References

High-dimensional multi-view learning via structural tensor estimation.

See Also

marm4

Examples

```
library(Mulste)
n <- 200; q <- 5; p <- 100; s <- 3; ng = 4
group <- rep(1:ng,each=p/ng)
mydata <- marm4.sim.fbs(n,q,p,s,group)
fit <- with(mydata, marm4.dr(Y,X,group,K,r1_index,r2_index,r3_index,r4_index,D0=D0,nlam=5))
```

---

marm4.sim.fbs *Generate scenario I data from structural MARM model.*

Description

Generate scenario I data for a structural multivariate additive model for multi-view data.

Usage

```
marm4.sim.fbs <- function(n,q,p,s,group=NULL,r10=2,r20=2,r30=2,r40=2,
                          isfixedR=0,D44=NULL,K=6,degr=3,sigma2=NULL,
                          seed_id=NULL,r1_index=NULL,r2_index=NULL,
                          r3_index=NULL,r4_index=NULL,D0=NULL)
```

Arguments

n	Sample size.
q	The number of responses, $q \geq 1$.
p	The number of covariates, $p \geq 1$.
s	The true covariates of each view associated with responses, $s \geq 1$.

group	A length-p vector of the grouping index of predictors, e.g., $group = c(1,1,1,2,2,2)$ means there are 6 perdictors in the model, and the first three perdictors are in the same group and the last three perdictors are in another one. By default, we set $group = c(rep(1, as.integer(p/2)), rep(2, p - as.integer(p/2))$.
r10	The first dimension of the tensor. Default is 2.
r20	The second dimension of the tensor. Default is 2.
r30	The third dimension of the tensor. Default is 2.
r40	The third dimension of the tensor. Default is 2.
isfixedR	A logical value indicating whether ranks are fixed.
D44	The mode of unfolding $D_{(4)}$. By default, D44 is generated by random.
K	The number of B-spline basis functions, that is the plus of both degrees of basis functioins and the number of knots. Default is 6, which means cubic splines.
degr	The number of knots of B-spline base function. Default is 3.
sigma2	err variance. Default is 0.1.
seed_id	A positive integer, the seed for generating the random numbers. Default is 1000.
r1_index	A user-specified sequence of r_1 values, where r_1 is the first dimension of single value matrix of the tensor. Default is r1_index= $1, \cdots, \min(\lceil \log(n) \rceil, p)$. if $isfixedR = 1$, r1_index is useless.
r2_index	A user-specified sequence of r_2 values, where r_2 is the second dimension of single value matrix of the tensor. Default is r2_index= $1, \cdots$,max{K_index}. if $isfixedR = 1$, r2_index is useless.
r3_index	A user-specified sequence of r_3 values, where r_3 is the third dimension of single value matrix of the tensor. Default is r3_index= $1, \cdots, \min(\lceil \log(n) \rceil, ng)$. if $isfixedR = 1$, r3_index is useless.
r4_index	A user-specified sequence of r_4 values, where r_4 is the third dimension of single value matrix of the tensor. Default is r4_index= $1, \cdots, \min(\lceil \log(n) \rceil, q)$. if $isfixedR = 1$, r4_index is useless.
D0	A user-specified list of initialized values, including five initialized matrices $S_{(4)}$ (called S), A, B, C and D. By default, a list of initialization satisfying fixed ranks is computed by random.

Details

This function can generate scenario I data of structural marm model. isfixedR are required to yield a different initialization D0. In scenario I, the true functions are exactly residing in the space of B-spline basis functions.

Value

Y	Response, a $n \times q$-matrix.
X	Design matrix, a $n \times p$-matrix.
f0	True functions, a $n \times p$-matrix.
group	The grouping index of predictors, a length-p vector.
D0	The initialized values.
...	Other options for algorithm.

References

High-dimensional multi-view learning via structural tensor estimation.

See Also

marm4.sim.fsin

Examples

```
library(Mulste)
n <- 200; q <- 5; p <- 100; s <- 3; ng = 4
group <- rep(1:ng,each=p/ng)
mydata <- marm4.sim.fbs(n,q,p,s,group)
```

`marm4.sim.fsin` *Generate scenario II data from structural MARM model.*

Description

Generate scenario II data for a structural multivariate additive model for multi-view data.

Usage

```
marm4.sim.fsin <- function(n,q,p,s,group=NULL,r10=2,r20=2,r30=2,r40=2,
                           isfixedR=0,D42=NULL,K=6,degr=3,sigma2=NULL,
                           seed_id=NULL,r1_index=NULL,r2_index=NULL,
                           r3_index=NULL,r4_index=NULL,D0=NULL)
```

Arguments

`n`	Sample size.
`q`	The number of responses, $q \geq 1$.
`p`	The number of covariates, $p \geq 1$.
`s`	The true covariates of each view associating to responses, $s \geq 1$.
`group`	A length-p vector of the grouping index of predictors, e.g., $group = c(1,1,1,2,2,2)$ means there are 6 perdictors in the model, and the first three perdictors are in the same group and the last three perdictors are in another one. By default, we set $group = c(rep(1, as.integer(p/2)), rep(2, p - as.integer(p/2))$.
`r10`	The first dimension of the tensor. Default is `2`.
`r20`	The second dimension of the tensor. Default is `2`.
`r30`	The third dimension of the tensor. Default is `2`.
`r40`	The third dimension of the tensor. Default is `2`.
`isfixedR`	A logical value indicating whether ranks are fixed.
`D42`	The mode of unfolding $D_{(2)}$. By default, `D2` is generated by random.
`K`	The number of B-spline basis functions, that is the plus of both degrees of basis functioins and the number of knots. Default is `6`, which means cubic splines.
`degr`	The number of knots of B-spline base function. Default is `3`.

sigma2	err variance. Default is 0.1.
seed_id	A positive integer, the seed for generating the random numbers. Default is 1000.
r1_index	A user-specified sequence of r_1 values, where r_1 is the first dimension of single value matrix of the tensor. Default is r1_index= $1, \cdots, \min(\lceil \log(n) \rceil, p)$. if $isfixedR = 1$, r1_index is useless.
r2_index	A user-specified sequence of r_2 values, where r_2 is the second dimension of single value matrix of the tensor. Default is r2_index= $1, \cdots$,max{K_index}. if $isfixedR = 1$, r2_index is useless.
r3_index	A user-specified sequence of r_3 values, where r_3 is the third dimension of single value matrix of the tensor. Default is r3_index= $1, \cdots, \min(\lceil \log(n) \rceil, ng)$. if $isfixedR = 1$, r3_index is useless.
r4_index	A user-specified sequence of r_4 values, where r_4 is the third dimension of single value matrix of the tensor. Default is r4_index= $1, \cdots, \min(\lceil \log(n) \rceil, q)$. if $isfixedR = 1$, r4_index is useless.
D0	A user-specified list of initialized values, including five initialized matrices $S_{(4)}$ (called S), A, B, C and D. By default, a list of initialization satisfying fixed ranks is computed by random.

Details

This function can generate scenario II data of structural marm model. isfixedR is required to yield a different initialization D0. In scenario II, the true functions are the linear combination of $sin(2\pi x)$ and $cos(\pi x)$.

Value

Y	Response, a $n \times q$-matrix.
X	Design matrix, a $n \times p$-matrix.
f0	True functions, a $n \times p$-matrix.
group	The grouping index of predictors, a length-p vector.
D0	The initialized values.
...	Other options for algorithm.

References

High-dimensional multi-view learning via structural tensor estimation.

See Also

marm4.sim.fbs

Examples

```
library(Mulste)
n <- 200; q <- 5; p <- 100; s <- 3; ng = 4
group <- rep(1:ng,each=p/ng)
mydata <- marm4.sim.fsin(n,q,p,s,group)
```

marmComposed *Fit composed model with sparsity assumption and fixed ranks.*

Description

Fit a composed model for multi-view data using B-splines with given ranks (r_{1g}, r_{2g}, r_{3g} and r_1, r_2, r_3, r_4). Multiple third-order coefficient tensors and a fourth-order coefficient tensor can be estimated by this function. The group sparse penalty such as LASSO, MCP or SCAD and the coordinate descent algorithm are used to yield a sparsity estimator. The BIC or cross-validation method are used to search the optimal regularization parameter.

Usage

```
marmComposed <-
 function(Y,X1,X2,G1=NULL,group=NULL,is.fabs=1,K=6,r31=NULL,r32=NULL,r33=NULL,r41=NULL,r42=NULL,
```

Arguments

Y	A $n \times q$ numeric matrix of responses.
X1	A $n \times p1$ numeric design matrix for the composed model, where $p1 = \sum_{1 \le g \le G1} p_g$
X2	A $n \times p2$ numeric design matrix for the composed model, where $p2 = \sum_{G1+1 \le g \le ng} p_g$ and ng is the number of views.
G1	The number of views that we do not consider their intergroup correlation.
group	A p vector of the grouping index of predictors, e.g., $group = c(1, 1, 1, 2, 2, 2)$ means there are 6 perdictors in the model, and the first three perdictors are in the same group and the last three perdictors are in another one. By default, we set $group = rep(1, p)$.
K	The number of B-spline basis functions, that is the plus of both degrees of basis functioins and the number of knots. Default is 6, which means cubic spline.
r31	The first dimension of single value matrix of the third-order tensor. Default is 2.
r32	The second dimension of single value matrix of the third-order tensor. Default is 2.
r33	The third dimension of single value matrix of the third-order tensor. Default is 2.
r41	The first dimension of single value matrix of the fourth-order tensor. Default is 2.
r42	The second dimension of single value matrix of the fourth-order tensor. Default is 2.
r43	The third dimension of single value matrix of the fourth-order tensor. Default is 2.
r44	The fourth dimension of single value matrix of the fourth-order tensor. Default is 2.
method	The method to be applied to select regularization parameters. Either BIC (default), or CV.
ncv	The number of cross-validation folds. Default is 10. If method is not CV, ncv is useless.

penalty	The penalty to be applied to the model. Either LASSO (the default), MCP or SCAD.
lambda	A user-specified sequence of lambda values. By default, a sequence of values of length nlam is computed, equally spaced on the log scale.
D0_t3	A user-specified list of initialized values for the third-order tensor, including ng sub-lists where ng is the number of groups. For each sub-list, it has four initialized matrix S_{(3)} (called S), A, B and C. By default, a list of initialization satisfying fixed ranks is computed by random.
D0_t4	A user-specified list of initialized values for the fourth-order tensor, including five initialized matrix $S_{(4)}$ (called S), A, B, C and D. By default, a list of initialization satisfying fixed ranks is computed by random.
intercept	A logical value indicating whether the intercept is fitted. Default is TRUE or set to zero by FALSE.
degr	The number of knots of B-spline base function. Default is 3.
nlam	The number of lambda values. Default is 20.
lam_min	The smallest value for lambda, as a fraction of lambda.max. Default is 0.01.
eps	Convergence threshhold. The algorithm iterates until the relative change in any coefficient is less than eps. Default is 1e-4.
max_step	Maximum number of iterations. Default is 20.
eps1	Convergence threshhold. The Coordinate descent method algorithm iterates until the relative change in any coefficient is less than eps1. Default is 1e-4.
max_step1_t3	The maximum iterates number of coordinate descent method during the estimation of the third-order tensor. Default is 20.
max_step1_t4	The maximum iterates number of coordinate descent method during the estimation of the fourth-order tensor. Default is 20.
gamma	The tuning parameter of the MCP/SCAD penalty.
dfmax1	Upper bound for the number of nonzero coefficients during the estimation of the third-order tensor. Default is no upper bound. However, for large data sets, computational burden may be heavy for models with a large number of nonzero coefficients.
dfmax2	Upper bound for the number of nonzero coefficients during the estimation of the fourth-order tensor. Default is no upper bound. However, for large data sets, computational burden may be heavy for models with a large number of nonzero coefficients.
alpha	Tuning parameter for the Mnet estimator which controls the relative contributions from the LASSO, MCP or SCAD penalty and the ridge, or L2 penalty. alpha=1 is equivalent to LASSO, MCP or SCAD penalty, while alpha=0 would be equivalent to ridge regression. However, alpha=0 is not supported; alpha may be arbitrarily small, but not exactly 0.
vnorm_ratio	The ratio between $lambda_1$ and $lambda_2$, that is, $lambda_2$=vnorm_ratio$\times lambda_1$. Default is 1.

Details

This function gives pq functional coefficients' estimators of composed model. Multiple third-order tensors with multiple ranks (r_{1g}, r_{2g}, r_{3g}) and a fourth-order tensor with multiple ranks (r_1, r_2, r_3, r_4) need to be estimated. We fix these ranks and use an alternative updating algorithm to update its core tensor and factor matrices based on Tucker decomposition. Group LASSO, SCAD or MCP penalty is applied on the row of each factor matrix A^g and A to achieve variable selection.

Value

D_t3	Estimator of coefficients corresponding to the third-order tensor $D_{(3)} = (D^1_{(3)}, ..., D^{G1}_{(3)})$.
D_t4	Estimator of coefficients corresponding to the fourth-order tensor $D_{(4)}$.
mu	Estimator of intercept μ.
S_t3.opt	A length-$G1$ list including estimator of the core tensor $S_{(3)}$ of each third-order coefficient tensor.
A_t3.opt	A length-$G1$ list including estimator of the factor matrix A of each third-order coefficient tensor.
B_t3.opt	A length-$G1$ list including estimator of the factor matrix B of each third-order coefficient tensor.
C_t3.opt	A length-$G1$ list including estimator of the factor matrix C of each third-order coefficient tensor.
S_t4.opt	A length-$ng - G1$ list including estimator of the core tensor $S_{(3)}$ of each fourth-order coefficient tensor.
A_t4.opt	A length-$ng - G1$ list including estimator of the factor matrix A of each fourth-order coefficient tensor.
B_t4.opt	A length-$ng - G1$ list including estimator of the factor matrix B of each fourth-order coefficient tensor.
C_t4.opt	A length-$ng - G1$ list including estimator of the factor matrix C of each fourth-order coefficient tensor.
D_t4.opt	A length-$ng - G1$ list including estimator of the factor matrix D of each fourth-order coefficient tensor.
lambda.seq	The sequence of regularization parameter values in the path.
lambda_opt	The value of lambda with the minimum BIC or CV value.
rss	Residual sum of squares (RSS).
df_t3	Degrees of freedom for all third-order tensors.
df_t4	Degrees of freedom for the fourth-order tensor.
activeX_t3	The active set of $X1$. A length-$p1$ vector.
activeX_t4	The active set of $X2$. A length-$p2$ vector.
opts	Other related parameters used in algorithm. Some of them are set by default.
opts_pen	Other related parameters used in algorithm (especially parameters in peanlty). Some of them are set by default.

References

High-dimensional multi-view learning via structural tensor estimation.

See Also

marmComposed.dr

Examples

```
library(Mulste)
n <- 200; q <- 5; p <- 100; s1 <- 5; s2 <- 3; G1 <- 1; ng <- 4
group <- rep(1:ng,each=p/ng)
mydata <- marmComposed.sim.fbs(n,q,p,s1,s2,G1,group,isfixedR=1)
fit <- with(mydata, marmComposed(Y,X1,X2,G1,group,is.fabs,K,r310,r320,r330,r410,r420,r430,r440,D0_t3=D0_t3,D
```

marmComposed.dr *Fit composed model with sparsity assumption and unknown ranks.*

Description

Fit a composed model using B-splines with unknown ranks (r_{1g}, r_{2g}, r_{3g} and r_1, r_2, r_3, r_4). Multiple third-order and fourth-order coefficient tensors can be estimated by this function. The group sparse penalty such as LASSO, MCP or SCAD and the coordinate descent algorithm are used to yield a sparsity estimator. The BIC or cross-validation method are used to search the optimal regularization parameter, multiple ranks and the number of B-spline basis functions simultaneously.

Usage

```
marmComposed.dr <-
 function(Y,X1,X2,G1=NULL,group=NULL,is.fabs=1,K_index=NULL,r_index=NULL,method="BIC",ncv=10,per
```

Arguments

Y	A $n \times q$ numeric matrix of responses.
X1	A $n \times p1$ numeric design matrix for the composed model, where $p1 = \sum_{1 \le g \le G1} p_g$
X2	A $n \times p2$ numeric design matrix for the composed model, where $p2 = \sum_{G1+1 \le g \le ng} p_g$ and ng is the number of views.
G1	The number of views that we do not consider their intergroup correlation.
group	A p vector of the grouping index of predictors, e.g., $group = c(1,1,1,2,2,2)$ means there are 6 perdictors in the model, and the first three perdictors are in the same group and the last three perdictors are in another one. By default, we set $group = rep(1, p)$.
is.fabs	A logical value indicating whether data comes from scenario I setting.
K_index	The user-specified sequence of K. Default is a length-1 vector 6.
r_index	A user-specified sequence of rank values for third-order and fourth-order tensors.
method	The method to be applied to select the number of B-spline basis functions, regularization parameters and multiple ranks simultaneously. Either BIC (default), or CV.
ncv	The number of cross-validation folds. Default is 10. If method is not CV, ncv is useless.
penalty	The penalty to be applied to the model. Either LASSO (the default), MCP or SCAD.
lambda	A user-specified sequence of lambda values. By default, a sequence of values of length nlam is computed, equally spaced on the log scale.
D0_t3	A user-specified list of initialized values for the third-order tensor, including ng sub-lists where ng is the number of groups. For each sub-list, it has four initialized matrix $S_{(3)}$ (called S), A, B and C. By default, a list of initialization satisfying fixed ranks is computed by random.
D0_t4	A user-specified list of initialized values for the fourth-order tensor, including five initialized matrix $S_{(4)}$ (called S), A, B, C and D. By default, a list of initialization satisfying fixed ranks is computed by random.

intercept	A logical value indicating whether the intercept is fitted. Default is TRUE or set to zero by FALSE.
degr	The number of knots of B-spline base function. Default is 3.
nlam	The number of lambda values. Default is 50.
lam_min	The smallest value for lambda, as a fraction of lambda.max. Default is 0.01.
eps	Convergence threshhold. The algorithm iterates until the relative change in any coefficient is less than eps. Default is 1e-4.
max_step	Maximum number of iterations. Default is 20.
eps1	Convergence threshhold. The Coordinate descent method algorithm iterates until the relative change in any coefficient is less than eps1. Default is 1e-4.
max_step1_t3	The maximum iterates number of coordinate descent method during the estimation of the third-order tensor. Default is 20.
max_step1_t4	The maximum iterates number of coordinate descent method during the estimation of the fourth-order tensor. Default is 20.
gamma	The tuning parameter of the MCP/SCAD penalty.
dfmax1	Upper bound for the number of nonzero coefficients during the estimation of the third-order tensor. Default is no upper bound. However, for large data sets, computational burden may be heavy for models with a large number of nonzero coefficients.
dfmax2	Upper bound for the number of nonzero coefficients during the estimation of the fourth-order tensor. Default is no upper bound. However, for large data sets, computational burden may be heavy for models with a large number of nonzero coefficients.
alpha	Tuning parameter for the Mnet estimator which controls the relative contributions from the LASSO, MCP or SCAD penalty and the ridge, or L2 penalty. alpha=1 is equivalent to LASSO, MCP or SCAD penalty, while alpha=0 would be equivalent to ridge regression. However, alpha=0 is not supported; alpha may be arbitrarily small, but not exactly 0.
vnorm_ratio	The ratio between $lambda_1$ and $lambda_2$, that is, $lambda_2$=vnorm_ratio$\times lambda_1$. Default is 1.

Details

This function gives pq functional coefficients' estimators of composed model. Multiple third-order tensors with unknown ranks (r_1, r_2, r_3) and a fourth-order tensor with unknown ranks (r_1, r_2, r_3, r_4) need to be estimated. The BIC or CV can be applied to select the optimal regularization parameters, multiple ranks and the number of B-spline basis functions simultaneously. An alternative updating algorithm can be used to update its core tensor and factor matrices based on Tucker decomposition. Group LASSO, SCAD or MCP penalty is applied on the row of each factor matrix A^g and A to achieve variable selection. Generally, the number of B-spline basis functions we need is fixed by 6, i.e., cubic splines are used to approximate the component functions.

Value

D_t3	Estimator of coefficients corresponding to the third-order tensor $D_{(3)} = (D^1_{(3)}, ..., D^{G1}_{(3)})$.
D_t4	Estimator of coefficients corresponding to the fourth-order tensor $D_{(4)}$.
mu	Estimator of intercept μ.

S_t3.opt	A length-$G1$ list including estimator of the core tensor $S_{(3)}$ of each third-order coefficient tensor.
A_t3.opt	A length-$G1$ list including estimator of the factor matrix A of each third-order coefficient tensor.
B_t3.opt	A length-$G1$ list including estimator of the factor matrix B of each third-order coefficient tensor.
C_t3.opt	A length-$G1$ list including estimator of the factor matrix C of each third-order coefficient tensor.
S_t4.opt	A length-$ng - G1$ list including estimator of the core tensor $S_{(3)}$ of each fourth-order coefficient tensor.
A_t4.opt	A length-$ng - G1$ list including estimator of the factor matrix A of each fourth-order coefficient tensor.
B_t4.opt	A length-$ng - G1$ list including estimator of the factor matrix B of each fourth-order coefficient tensor.
C_t4.opt	A length-$ng - G1$ list including estimator of the factor matrix C of each fourth-order coefficient tensor.
D_t4.opt	A length-$ng - G1$ list including estimator of the factor matrix D of each fourth-order coefficient tensor.
rk_t3_opt	The optimal ranks and the number of B-spline basis functions for the third-order tensor that slected by BIC, or CV. It is a vector with length 4, which are selected r_1, r_2, r_3, and K.
rk_t4_opt	The optimal ranks and the number of B-spline basis functions for the fourth-order tensor that selected by BIC, or CV. It is a vector with length 5, which are selected r_1, r_2, r_3, r_4 and K.
lambda.seq	The sequence of regularization parameter values in the path.
lambda_opt	The value of lambda with the minimum BIC or CV value.
rss	Residual sum of squares (RSS).
df_t3	Degrees of freedom for all third-order tensors.
df_t4	Degrees of freedom for the fourth-order tensor.
activeX_t3	The active set of $X1$. A length-$p1$ vector.
activeX_t4	The active set of $X2$. A length-$p2$ vector.
opts	Other related parameters used in algorithm. Some of them are set by default.
opts_pen	Other related parameters used in algorithm (especially parameters in peanlty). Some of them are set by default.

References

High-dimensional multi-view learning via structural tensor estimation.

See Also

marmComposed

Examples

```
library(Mulste)
n <- 200; q <- 5; p <- 100; s1 <- 5; s2 <- 3; G1 <- 1; ng = 4
group <- rep(1:ng,each=p/ng)
mydata <- marmComposed.sim.fbs(n,q,p,s1,s2,G1,group)
fit <- with(mydata, marmComposed.dr(Y,X1,X2,G1,group,is.fabs,K,r_index,D0_t3=D0_t3,D0_t4=D0_t4,nlam=5))
```

`marmComposed.sim.fbs` *Generate scenario I data from composed model.*

Description

Generate scenario I data for composed model.

Usage

```
marmComposed.sim.fbs <- function(n,q,p,s1,s2,G1=NULL,group=NULL,r3_10=2,r3_20=2,r3_30=2,r4_10=2,r
```

Arguments

`n`	Sample size.
`q`	The number of responses, $q \geq 1$.
`p`	The number of covariates, $p \geq 1$.
`s1`	The true covariates of each first G1 views associated with responses, $s1 \geq 1$.
`s2`	The true covariates of each first G1 views associated with responses, $s2 \geq 1$.
`G1`	The number of views that we do not consider their intergroup correlation.
`group`	A length-p vector of the grouping index of predictors, e.g., $group = c(1,1,1,2,2,2)$ means there are 6 perdictors in the model, and the first three perdictors are in the same group and the last three perdictors are in another one.
`r310`	The first dimension of the third-order tensor. Default is `2`.
`r320`	The second dimension of the third-order tensor. Default is `2`.
`r330`	The third dimension of the third-order tensor. Default is `2`.
`r410`	The first dimension of the fourth-order tensor. Default is `2`.
`r420`	The second dimension of the fourth-order tensor. Default is `2`.
`r430`	The third dimension of the fourth-order tensor. Default is `2`.
`r440`	The third dimension of the fourth-order tensor. Default is `2`.
`isfixedR`	A logical value indicating whether ranks are fixed.
`D3`	The mode of unfolding $D_{(3)}$. By default, `D3` is generated by random.
`D44`	The mode of unfolding $D_{(4)}$. By default, `D44` is generated by random.
`K`	The number of B-spline basis functions, that is the plus of both degrees of basis functioins and the number of knots. Default is `6`, which means cubic splines.
`degr`	The number of knots of B-spline base function. Default is `3`.
`sigma2`	err variance. Default is `0.1`.
`seed_id`	A positive integer, the seed for generating the random numbers. Default is `1000`.
`r1_t3_index`	A user-specified sequence of r_1 values, where r_1 is the first dimension of single value matrix of the third-order tensor. Default is `r31_index`$= 1, \cdots, \min(\lceil \log(n) \rceil, p)$. if $isfixedR = 1$, `r31_index` is useless.
`r2_t3_index`	A user-specified sequence of r_2 values, where r_2 is the second dimension of single value matrix of the third-order tensor. Default is `r32_index`$= 1, \cdots$,`max{K_index}`. if $isfixedR = 1$, `r2_index` is useless.

r3_t3_index	A user-specified sequence of r_3 values, where r_3 is the third dimension of single value matrix of the third-order tensor. Default is r33_index$= 1, \cdots, \min(\lceil \log(n) \rceil, q)$. if $isfixedR = 1$, r3_index is useless.
r1_t4_index	A user-specified sequence of r_1 values, where r_1 is the first dimension of single value matrix of the fourth-order tensor. Default is r41_index$= 1, \cdots, \min(\lceil \log(n) \rceil, p)$. if $isfixedR = 1$, r1_index is useless.
r2_t4_index	A user-specified sequence of r_2 values, where r_2 is the second dimension of single value matrix of the fourth-order tensor. Default is r42_index$= 1, \cdots$,max{K_index}. if $isfixedR = 1$, r2_index is useless.
r3_t4_index	A user-specified sequence of r_3 values, where r_3 is the third dimension of single value matrix of the fourth-order tensor. Default is r43_index$= 1, \cdots, \min(\lceil \log(n) \rceil, ng)$. if $isfixedR = 1$, r3_index is useless.
r4_t4_index	A user-specified sequence of r_4 values, where r_4 is the third dimension of single value matrix of the fourth-order tensor. Default is r44_index$= 1, \cdots, \min(\lceil \log(n) \rceil, q)$. if $isfixedR = 1$, r4_index is useless.
D0_t3	A user-specified list of initialized values for the third-order tensor, including ng sub-lists where ng is the number of groups. For each sub-list, it has four initialized matrices $S_{(3)}$ (called S), A, B and C. By default, a list of initialization satisfying fixed ranks is computed by random.
D0_t4	A user-specified list of initialized values for the fourth-order tensor, including five initialized matrices $S_{(4)}$ (called S), A, B, C and D. By default, a list of initialization satisfying fixed ranks is computed by random.

Details

This function can generate scenario I data of composed model. isfixedR is required to yield a different initialization D0_t3 and D0_t4. In scenario I, the true functions are exactly residing in the space of B-spline basis functions.

Value

Y	Response, a $n \times q$-matrix.
X1	Design matrix, a $n \times p1$-matrix where p1 is a the whole number of covariates in the first G1 views.
X2	Design matrix, a $n \times p2$-matrix where p2 is a the whole number of covariates in the last ng-G1 views. ng is the number of views.
f01	True functions, a $n \times p1$-matrix.
f02	True functions, a $n \times p2$-matrix.
group	The grouping index of predictors, a length-p vector.
D0_t3	The initialized values for the third-order tensor.
D0_t4	The initialized values for the fourth-order tensor.
...	Other options for algorithm.

References

High-dimensional multi-view learning via structural tensor estimation.

See Also

marmComposed.sim.fsin

Examples

```
library(Mulste)
n <- 200; q <- 5; p <- 100; s1 <- 5; s2 <- 3; G1 <- 1; ng = 4
group <- rep(1:ng,each=p/ng)
mydata <- marmComposed.sim.fbs(n,q,p,s1,s2,G1,group)
```

`marmComposed.sim.fsin` *Generate scenario II data from composed model.*

Description

Generate scenario II data for a multivariate additive model for multi-view data.

Usage

```
marmComposed.sim.fsin <- function(n,q,p,s1,s2,G1=NULL,group=NULL,r3_10=2,r3_20=2,r3_30=2,r4_10=2,
```

Arguments

n	Sample size.
q	The number of responses, $q \geq 1$.
p	The number of covariates, $p \geq 1$.
s1	The true covariates of each first G1 views associated with responses, $s1 \geq 1$.
s2	The true covariates of each first G1 views associated with responses, $s2 \geq 1$.
G1	The number of views that we do not consider their intergroup correlation.
group	A length-p vector of the grouping index of predictors, e.g., $group = c(1,1,1,2,2,2)$ means there are 6 perdictors in the model, and the first three perdictors are in the same group and the last three perdictors are in another one. By default, we set $group = rep(1,p)$.
r310	The first dimension of the third-order tensor. Default is 2.
r320	The second dimension of the third-order tensor. Default is 2.
r330	The third dimension of the third-order tensor. Default is 2.
r410	The first dimension of the fourth-order tensor. Default is 2.
r420	The second dimension of the fourth-order tensor. Default is 2.
r430	The third dimension of the fourth-order tensor. Default is 2.
r440	The third dimension of the fourth-order tensor. Default is 2.
isfixedR	A logical value indicating whether ranks are fixed.
D2	The mode of unfolding $D_{(2)}$. By default, D2 is generated by random.
D42	The mode of unfolding $D_{(4)}$. By default, D42 is generated by random.
K	The number of B-spline basis functions, that is the plus of both degrees of basis functioins and the number of knots. Default is 6, which means cubic spline.
degr	The number of knots of B-spline base function. Default is 3.
sigma2	err variance. Default is 0.1.
seed_id	A positive integer, the seed for generating the random numbers. Default is 1000.

r1_t3_index	A user-specified sequence of r_1 values, where r_1 is the first dimension of single value matrix of the third-order tensor. Default is r31_index$= 1, \cdots, \min(\lceil \log(n) \rceil, p)$. if $isfixedR = 1$, r31_index is useless.
r2_t3_index	A user-specified sequence of r_2 values, where r_2 is the second dimension of single value matrix of the third-order tensor. Default is r32_index$= 1, \cdots$,max{K_index}. if $isfixedR = 1$, r2_index is useless.
r3_t3_index	A user-specified sequence of r_3 values, where r_3 is the third dimension of single value matrix of the third-order tensor. Default is r33_index$= 1, \cdots, \min(\lceil \log(n) \rceil, q)$. if $isfixedR = 1$, r3_index is useless.
r1_t4_index	A user-specified sequence of r_1 values, where r_1 is the first dimension of single value matrix of the fourth-order tensor. Default is r41_index$= 1, \cdots, \min(\lceil \log(n) \rceil, p)$. if $isfixedR = 1$, r1_index is useless.
r2_t4_index	A user-specified sequence of r_2 values, where r_2 is the second dimension of single value matrix of the fourth-order tensor. Default is r42_index$= 1, \cdots$,max{K_index}. if $isfixedR = 1$, r2_index is useless.
r3_t4_index	A user-specified sequence of r_3 values, where r_3 is the third dimension of single value matrix of the fourth-order tensor. Default is r43_index$= 1, \cdots, \min(\lceil \log(n) \rceil, ng)$. if $isfixedR = 1$, r3_index is useless.
r4_t4_index	A user-specified sequence of r_4 values, where r_4 is the third dimension of single value matrix of the fourth-order tensor. Default is r44_index$= 1, \cdots, \min(\lceil \log(n) \rceil, q)$. if $isfixedR = 1$, r4_index is useless.
D0_t3	A user-specified list of initialized values for the third-order tensor, including ng sub-lists where ng is the number of groups. For each sub-list, it has four initialized matrices $S_{(3)}$ (called S), A, B and C. By default, a list of initialization satisfying fixed ranks is computed by random.
D0_t4	A user-specified list of initialized values for the fourth-order tensor, including five initialized matrices $S_{(4)}$ (called S), A, B, C and D. By default, a list of initialization satisfying fixed ranks is computed by random.

Details

This function can generate scenario II data of composed model. isfixedR is required to yield a different initialization D0_t3 and D0_t4. In scenario II, the true functions are the linear combination of $sin(2\pi x)$ and $cos(\pi x)$.

Value

Y	Response, a $n \times q$-matrix.
X1	Design matrix, a $n \times p1$-matrix where p1 is a the whole number of covariates in the first G1 views.
X2	Design matrix, a $n \times p2$-matrix where p2 is a the whole number of covariates in the last ng-G1 views. ng is the number of views.
f01	True functions, a $n \times p1$-matrix.
f02	True functions, a $n \times p2$-matrix.
group	The grouping index of predictors, a length-p vector.
D0_t3	The initialized values for the third-order tensor.
D0_t4	The initialized values for the fourth-order tensor.
...	Other options for algorithm.

References

High-dimensional multi-view learning via structural tensor estimation.

See Also

marmComposed.sim.fbs

Examples

```
library(Mulste)
n <- 200; q <- 5; p <- 100; s1 <- 5; s2 <- 3; G1 <- 1; ng = 4
group <- rep(1:ng,each=p/ng)
mydata <- marmComposed.sim.fsin(n,q,p,s1,s2,G1,group)
```

Index