Package ‘tensorMAM’

March 17, 2020

Type Package

Title tensorMAM

Version 0.1.0

Author Xu Liu [aut,cre],
Jian Huang [aut],
Heng Lian [aut],
Xiangyong Tan [ctb]

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

Description A tensor Estimation approach to multivariate additive models. The B-splines are used to approximate unknown function. The number of predictors can be diverged as sample size increases, in which the penalty LASSO, MCP or SCAD can be used.

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/tensorMAM

Encoding UTF-8

R topics documented:

`tensorMAM-package` *A tensor estimation approach to multivariate additive models*

Description

For a high-dimensional multivariate additive model (MAM) using B-splines, with or without aparsity assumptions, treating the coefficients as a third-order tensor and borrowing Tucker decomposition to reduce the number of parameters. The multivariate sparse group lasso (mcp or scad) and the coordinate descent algorithm are used to estimate functions for sparsity situation.

Details

This section should provide a more detailed overview of how to use the package, including the most important functions.

Author(s)

Xu Liu

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

References

A tensor estimation approach to multivariate additive models.

`breastData` *Breast cancer gene expression and DNA copy number dataset*

Description

The breast cancer dataset includes gene expressions and comparative genomic hybridization measurements for 89 subjects, which is from Chin et al. (2006). This dataset has been considered by Witten et al. (2009) and Chen et al. (2013). In our paper, we selected chromosome 21, including $q = 44$ variables for copy-number variations and $p = 227$ variables for gene expression. As in Chen et al. (2013), we consider copy-number variations as the responses and gene expressions as the predictors.

Usage

```
data(breastData)
```

Details

The "breastData" is formated as a list with elements:

dna:	the CGH spots , a matrix with size 2149×89 and the smaple size 89
rna:	genes, a matrix with size 19672×89 and the smaple size 89
chrom:	chromosomal location of each CGH spot, a 2149-vector
nuc:	nucleotide position of each CGH spot, a 2149-vector
gene:	an accession number for each gene, a 19672-vector
genenames:	gene name, a 19672-vector

genechr:	chromosomal location of each gene, a 19672-vector
genedesc:	description of each gene, a 19672-vector
genepos:	nucleotide position of each gene, a 19672-vector

References

Chin, K., DeVries, S., Fridlyand, J., Spellman, P., Roydasgupta, R., Kuo,W.-L., Lapuk, A., Neve, R., Qian, Z., Ryder, T., Chen, F., Feiler, H., Tokuyasu, T., Kingsley, C., Dairkee, S., Meng, Z., Chew, K., Pinkel, D., Jain, A., Ljung, B., Esserman, L., Albertson, D.,Waldman, F. & Gray, J. (2006). Genomic and transcriptional aberrations linked to breast cancer pathophysiologies. Cancer cell **10 (6)**, 429-541.

Witten, D. M., Tibshirani, R. and Hastie, T. (2009). A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. Biostatistics **10 (3)**, 515-534.

Chen, K., Dong, H., and Chan, K. S. (2013). Reduced rank regression via adaptive nuclear norm penalization. Biometrika, **100 (4)**, 901-920.

Examples

```
data(breastData)
attach(breastData)
Y = t(dna[chrom==21,])
Xt = t(rna[which(genechr==21),])
n = nrow(Y)

minX = apply(Xt,2,min)
maxX = apply(Xt,2,max)
X = (Xt - matrix(rep(minX,each = n),n))/matrix(rep(maxX-minX,each = n),n)
Y = scale(Y)
fit <- mam_sparse_dr(Y[,1:5], X[,1:10])
D3hat <- fit$Dnew
opt <- fit$rk_opt
detach(breastData)
```

`generateData` *Generate data from MAM model.*

Description

Generate data for a high-dimensional multivariate additive model, with or without aparsity assumptions.

Usage

```
generateData(n, q, p, s, D2, sigma2=NULL, indexF=NULL, seed_id=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 associating to response, $s \geq 1$.
D2	The mode of unfolding $D_{(2)}$.
sigma2	err variance. Default is 0.1.
indexF	A $q \times s$ matrix. The index of significant predictors corresponding to response y_l. Default is the matrix with each row being $(1, 2, \cdots, s)$.
seed_id	A positive integer, the seed for generating the random numbers.

Details

This function gives pq functional coefficients' estimators of MAM. The singular value matrix of tensor is a $r_1 \times r_2 \times r_3$-tensor. We choose r_1, r_2 and r_3 by BIC, AIC, EBIC, CV, or GCV.

Value

Y	Response, a $n \times q$-matrix.
X	Design matrix, a $n \times p$-matrix.
f0	True functions.

References

A tensor estimation approach to multivariate additive models.

See Also

mam_sparse

Examples

```
# Example 1

D2 <- matrix(runif(30, 0.7, 1), 2, 15)
mydata <- generateData(200, 3, 5, 5, D2)

Y <- mydata$Y
X <- mydata$X


# Example 2
n <- 500
p <- 10
q <- 10
s <- 10
K <- 6
s0 <- s
r10=r20=r30=2
S3 <- matrix(runif(r10*r20*r30,3,7),nrow = r30)
T1 <- matrix(rnorm(s0*r10),nrow = s0)
U1 <- qr.Q(qr(T1))
```

```
T1 <- matrix(rnorm(K*r20),nrow = K)
U2 <- qr.Q(qr(T1))
T1 <- matrix(rnorm(q*r30),nrow = q)
U3 <- qr.Q(qr(T1))
D3 <- U3%*%S3%*%t(kronecker(U2,U1))
D2 <- TransferModalUnfoldings(D3,3,2,s0,K,q)
mydata <- generateData(n,q,p,s0,D2)
```

`mam` *Fit MAM without sparsity assumption and with fixed ranks.*

Description

Fit a low-dimensional multivariate additive model using B-splines, without aparsity assumptions, and given ranks r_1, r_2, r_3.

Usage

```
mam(Y, X, r1 = NULL, r2 = NULL, r3 = NULL, SABC = NULL,
          intercept = TRUE, K = 6, degr = 3, eps = 1e-4, max_step = 20)
```

Arguments

`Y`	A $n \times q$ numeric matrix of responses.
`X`	A $n \times p$ numeric design matrix for the model.
`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.
`SABC`	A user-specified list of initial coefficient matrix of S, A, B, C. By default, initial matrices are provided by random.
`intercept`	Should intercept(s) be fitted (default=`TRUE`) or set to zero (`FALSE`)?
`K`	The number of B-spline base function, that is the plus of both degrees of base functioin and the number of knots. Default is `6`.
`degr`	the number of knots of B-spline base function. Default is `3`.
`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.

Details

This function gives `pq` functional coefficients' estimators of MAM. The singular value matrix of tensor is a $r_1 \times r_2 \times r_3$-tensor. We choose r_1, r_2 and r_3 by `BIC` or `CV`.

Value

`Dnew`	Estimator of $D_{(3)}$.
`mu`	Estimator of intercept μ.
`rss`	Residual sum of squares (RSS).
`Y`	Response Y.
`X`	Design matrix X.
`Z`	Design matrix of Bspline approximation.

References

A tensor estimation approach to multivariate additive models.

See Also

mam_sparse

Examples

```
p <- 5
q <- 5
D2 <- matrix(runif(2*p*q, 0.7, 1), 2, p*q) # tensor with size 5*2*5
mydata <- generateData(200, q, p, p, D2)

fit <- mam(mydata$Y, mydata$X)
K <- fit$K
D3hat <- fit$Dnew # A q*(Kp) matrix with (p,K,q)=(5,6,5)
D2hat <- TransferModalUnfoldings(D3hat,3,2,p,K,q)
```

---

mam_dr   *Fit MAM without sparsity assumption, and with ranks selected by* BIC, AIC, EBIC, CV, *or* GCV.

Description

Fit a low-dimensional multivariate additive model using B-splines, without aparsity assumptions, and with ranks r_1, r_2, r_3 selected by BIC, AIC, EBIC, CV, or GCV.

Usage

```
mam_dr(Y, X, criteria = "BIC", ncv = 10, r1_index = NULL, r2_index = NULL,
       r3_index = NULL, SABC = NULL, intercept = TRUE, K = 6, degr = 3,
       eps = 1e-4, max_step = 20)
```

Arguments

Y	A $n \times q$ numeric matrix of responses.
X	A $n \times p$ numeric design matrix for the model.
criteria	The criteria to be applied to select parameters. Either BIC (the default), AIC, EBIC, CV, or GCV.
ncv	The number of cross-validation folds. Default is 10. If criteria is not "CV", ncv is useless.
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= $1, \cdots, \min(\log(n)\rceil, p)$.
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= $1, \cdots, \min(\log(n)\rceil, q)$.
SABC	A user-specified list of initial coefficient matrix of S, A, B, C, which is a list with values S, A, B, C. By default, initial matrices are provided by random.

`intercept`	Should intercept(s) be fitted (default=`TRUE`) or set to zero (`FALSE`)?
`K`	The number of B-spline base function, that is the plus of both degrees of base functioin and the number of knots. Default is `6`.
`degr`	the number of knots of B-spline base function. Default is `3`.
`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.

Details

This function gives `pq` functional coefficients' estimators of MAM. The singular value matrix of tensor is a $r_1 \times r_2 \times r_3$-tensor. We choose r_1, r_2 and r_3 by `BIC` or `CV`.

Value

`Dnew`	Estimator of $D_{(3)}$.
`mu`	Estimator of intercept μ.
`rss`	Residual sum of squares (RSS).
`rk_opt`	The optimal parametres that slected by `BIC`, `AIC`, `EBIC`, `CV`, or `GCV`. It is a vector with length 4, which are selected r_1, r_2, r_3, and K.
`selected`	Which λ is selection.
`Y`	Response Y.
`X`	Design matrix X.
`Z`	Design matrix of Bspline approximation.

References

A tensor estimation approach to multivariate additive models.

See Also

mam, mam_sparse_dr

Examples

```
p <- 5
q <- 5
D2 <- matrix(runif(2*p*q, 0.7, 1), 2, p*q) # tensor with size 5*2*5
mydata <- generateData(200, q, p, p, D2)

fit <- mam_dr(mydata$Y, mydata$X)
K <- fit$K
D3hat <- fit$Dnew  # A q*(Kp) matrix with (p,K,q)=(5,6,5)
D2hat <- TransferModalUnfoldings(D3hat,3,2,p,K,q)
opt <- fit$rk_opt
```

`mam_sparse` *Fit MAM with sparsity assumption and fixed ranks.*

Description

Fit a high-dimensional multivariate additive model using B-splines, with or without aparsity assumptions, and given ranks r_1, r_2, r_3. The multivariate sparse group LASSO, MCP or SCAD) and the coordinate descent algorithm are used to estimate functions for sparsity situation.

Usage

```
mam_sparse(Y, X, criteria="BIC",ncv=10, r1 = NULL, r2 = NULL, r3 = NULL,
           penalty="LASSO",  isPenColumn=TRUE, lambda = NULL, SABC = NULL,
           intercept = TRUE, initMethod="LASSO", K = 6, degr = 3, nlam = 20,
           lam_min = 1e-3, eps1 = 1e-4, maxstep1 = 20, eps2 = 1e-4,
           maxstep2 = 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.
criteria	The criteria to be applied to select parameters. Either BIC (the default), AIC, EBIC, CV, or GCV.
ncv	The number of cross-validation folds. Default is 10. If criteria is not CV, ncv is useless.
penalty	The penalty to be applied to the model. Either LASSO (the default), MCP or SCAD.
isPenColumn	A logical value indicating whether the coefficients associating with X_j that affects whole response y is penalized. Default is TRUE. If isPenColumn is TRUE, the coefficients associating with X_j that affects simultaneously whole response y is penalized for each $j \in \{1, \cdots, p\}$. If isPenColumn is FALSE, the coefficients associating with X_j that affects single response y_l is penalized for each $j \in \{1, \cdots, p\}$, where $l \in \{1, \cdots, q\}$.
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.
SABC	A user-specified list of initial coefficient matrix of S, A, B, C. By default, initial matrices are provided by random.
intercept	Should intercept(s) be fitted (default=TRUE) or set to zero (FALSE)?
initMethod	One can estimate the initial tensor $D_{(3)}$ as a metrix by choosing a penalty to penalize group-column wise. initcriteria can be LASSO, MCP or SCAD. The default is LASSO
K	The number of B-spline base function, that is the plus of both degrees of base functioin and the number of knots. Default is 6.
degr	The number of knots of B-spline base function. Default is 3.
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.

nlam	The number of lambda values. Default is 20.
lam_min	The smallest value for lambda, as a fraction of lambda.max. Default is 1e-3.
eps1	Convergence threshhold. The algorithm iterates until the relative change in any coefficient is less than eps1. Default is 1e-4.
maxstep1	Maximum number of iterations. Default is 20.
eps2	Convergence threshhold. The Coordinate descent method algorithm iterates until the relative change in any coefficient is less than eps2. Default is 1e-4.
maxstep2	The maximum iterates number of coordinate descent method. Default is 20.
gamma	The tuning parameter of the MCP/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/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 MAM. The singular value matrix of tensor is a $r_1 \times r_2 \times r_3$-tensor. r_1, r_2 and r_3 are fixed.

Value

betapath	Solution path of β.
rss	Residual sum of squares (RSS).
df	Degrees of freedom.
lambda	The sequence of regularization parameter values in the path.
lambda_opt	The value of lambda with the minimum BIC value.
selectedID	The index of lambda corresponding to lambda_opt.
activeX	The active set of X. It is a p dimensional vector.
activeF	The active set of functions. It is a $q \times p$ matrix.
Dnew	Estimator of $D_{(3)}$.
mu	Estimator of intercept μ.
Y	Response Y.
X	Design matrix X.
Z	Design matrix of Bspline approximation λ.

References

A tensor estimation approach to multivariate additive models.

See Also

mam, mam_sparse_dr

Examples

```
p <- 10
q <- 5
s <- 5
D2 <- matrix(runif(2*s*q, 0.7, 1), 2, s*q) # tensor with size 5*2*5
mydata <- generateData(200, q, p, s, D2)

fit <- mam_sparse(mydata$Y, mydata$X)
K <- fit$K
D3hat <- fit$Dnew  # A q*(Kp) matrix with (p,K,q)=(10,6,5)
D2hat <- TransferModalUnfoldings(D3hat,3,2,p,K,q)
D1hat <- TransferModalUnfoldings(D3hat,3,1,p,K,q)
which(rowSums(D1hat^2)>0)
fit$activeX
```

`mam_sparse_dr` *Fit MAM with sparsity assumption and ranks selected by* `BIC`*,* `AIC`*,* `EBIC`*,* `CV`*, or* `GCV`*.*

Description

Fit a high-dimensional multivariate additive model using B-splines, with or with aparsity assumptions and ranks selected by `BIC`, `AIC`, `EBIC`, `CV`, or `GCV`. The multivariate sparse group `LASSO`, `MCP` or `SCAD`) and the coordinate descent algorithm are used to estimate functions for sparsity situation. The tuning parameter is selected by `BIC`, `AIC`, `EBIC`, `CV`, or `GCV`, which matchs the method of rank selection.

Usage

```
mam_sparse_dr(Y, X, criteria = "BIC", ncv = 10, penalty = "LASSO", isPenColumn=TRUE,
              r1_index = NULL, r2_index = NULL, r3_index = NULL, lambda = NULL,
              SABC = NULL, intercept = TRUE, initMethod="LASSO", nlam = 50,
              K = 6, degr = 3, lam_min = 0.01, eps1 = 1e-4, maxstep1 = 20,
              eps2 = 1e-4, maxstep2 = 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.
`criteria`	The criteria to be applied to select parameters. Either `BIC` (the default), `AIC`, `EBIC`, `CV`, or `GCV`.
`ncv`	The number of cross-validation folds. Default is 10. If `criteria` is not `CV`, `ncv` is useless.
`penalty`	The penalty to be applied to the model. Either `LASSO` (the default), `MCP` or `SCAD`.
`isPenColumn`	A logical value indicating whether the coefficients associating with X_j that affects whole response y is penalized. Default is `TRUE`. If `isPenColumn` is `TRUE`, the coefficients associating with X_j that affects simultaneously whole response y is penalized for each $j \in \{1, \cdots, p\}$. If `isPenColumn` is `FALSE`, the coefficients associating with X_j that affects single response y_l is penalized for each $j \in \{1, \cdots, p\}$, where $l \in \{1, \cdots, q\}$.

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= $1, \cdots, \min(\lceil \log(n) \rceil, q)$.
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.
SABC	A user-specified list of initial coefficient matrix of S, A, B, C. By default, initial matrices are provided by random.
intercept	Should intercept(s) be fitted (default=TRUE) or set to zero (FALSE)?
initMethod	One can estimate the initial tensor $D_{(3)}$ as a metrix by choosing a penalty to penalize group-column wise. initMethod can be LASSO, MCP or SCAD. The default is LASSO
nlam	The number of lambda values. Default is 50.
K	The number of B-spline base function, that is the plus of both degrees of base functioin and the number of knots. Default is 6.
degr	The number of knots of B-spline base function. Default is degr = 3.
lam_min	The smallest value for lambda, as a fraction of lambda.max. Default is 1e-2.
eps1	Convergence threshhold. The algorithm iterates until the relative change in any coefficient is less than eps1. Default is 1e-4.
maxstep1	Maximum number of iterations. Default is 20.
eps2	Convergence threshhold. The Coordinate descent method algorithm iterates until the relative change in any coefficient is less than eps2. Default is 1e-4.
maxstep2	The maximum iterates number of coordinate descent method. Default is 20.
gamma	The tuning parameter of the MCP/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/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 MAM. The singular value matrix of tensor is a $r_1 \times r_2 \times r_3$-tensor. We choose r_1, r_2 and r_3 by BIC or CV.

Value

Dnew	Estimator of $D_{(3)}$.
mu	Estimator of intercept μ.
rss	Residual sum of squares (RSS).
df	Degrees of freedom.

activeX	The active set of X. It is a p dimensional vector.
activeF	The active set of functions. It is a $q \times p$ matrix.
lambda	The sequence of regularization parameter values in the path.
selectedID	The index of lambda corresponding to lambda_opt.
lambda_opt	The value of lambda with the minimum BIC, AIC, EBIC, CV, or GCV value.
RSS	The values of BIC or CV, which is a vector.
rk_opt	The optimal parametres that slected by BIC, AIC, EBIC, CV, or GCV. It is a vector with length 4, which are selected r_1, r_2, r_3, and K.
Y	Response Y.
X	Design matrix X.
Z	Design matrix of Bspline approximation.

References

A tensor estimation approach to multivariate additive models.

See Also

mam_dr, mam_sparse

Examples

```
#Example 1

p <- 10
q <- 5
s <- 5
D2 <- matrix(runif(2*s*q, 0.7, 1), 2, s*q) # tensor with size 5*2*5
mydata <- generateData(200, q, p, s, D2)

fit <- mam_sparse_dr(mydata$Y, mydata$X)
K <- fit$K
D3hat <- fit$Dnew  # A q*(Kp) matrix with (p,K,q)=(5,6,5)
D2hat <- TransferModalUnfoldings(D3hat,3,2,p,K,q)
D1hat <- TransferModalUnfoldings(D3hat,3,1,p,K,q)
opt <- fit$rk_opt
which(rowSums(D1hat^2)>0)
fit$activeX

#Example 2

data(breastData)
attach(breastData)
Y = t(dna[chrom==21,])
Xt = t(rna[which(genechr==21),])
n = nrow(Y)

minX = apply(Xt,2,min)
maxX = apply(Xt,2,max)
X = (Xt - matrix(rep(minX,each = n),n))/matrix(rep(maxX-minX,each = n),n)
Y = scale(Y)
fit <- mam_sparse_dr(Y[,1:5], X[,1:10])
K <- fit$K
```

```
D3hat <- fit$Dnew  # A q*(Kp) matrix with (p,K,q)=(10,6,5)
D1hat <- TransferModalUnfoldings(D3hat,3,1,10,K,5)
opt <- fit$rk_opt
which(rowSums(D1hat^2)>0)
fit$activeX
detach(breastData)
```

`mvrblockwise` *Estimate coefficients of high-dimensional multivariate regression for the grouped-column-wise*

Description

This function provides the coefficient matrix estimator of high-dimensional multivariate regression (MVR) with penalty `LASSO`, `MCP` or `SCAD`). The tuning parameter is selected by `BIC` (the default), `AIC`, `EBIC`, `CV`, or `GCV`.

Usage

```
mvrblockwise(Y, X ,Z=NULL, criteria="BIC", ncv=10, penalty="LASSO",
             isPenColumn=TRUE, group=NULL, lambda=NULL, nlam=50,
             intercept=TRUE, lam_min=1e-4, eps=1e-6, max_step=50,
             gamma_pen=2, dfmax=NULL, alpha=1)
```

Arguments

`Y`	The response, a vector of size n or a matrix of size $n \times q$.
`X`	The covariates to be penalized, a matrix with dimension $n \times p$.
`Z`	The covariates without penalization, a matrix with dimension $n \times d$. The default is `NULL`.
`criteria`	The criteria to be applied to select parameters. Either `BIC` (the default), `AIC`, `EBIC`, `CV`, or `GCV`.
`ncv`	The number of cross-validation folds. Default is 10. If `criteria` is not `CV`, `ncv` is useless.
`penalty`	The penalty to be applied to the model. Either `LASSO` (the default), `MCP` or `SCAD`.
`isPenColumn`	A logical value indicating whether the coefficients associating with X_j that affects whole response Y is penalized. Default is `TRUE`. If `isPenColumn` is `TRUE`, the coefficients associating with X_j that affects simultaneously whole response y is penalized for each $j \in \{1, \cdots, p\}$. If `isPenColumn` is `FALSE`, the coefficients associating with X_j that affects single response Y_l is penalized for each $j \in \{1, \cdots, p\}$, where $l \in \{1, \cdots, q\}$.
`group`	A integer vector describing the grouping of the coefficients. For example, we can preset `group = rep(1:G,each=K)`. If no grouping, `group = rep(1:ncol(X))`. The default is `group = rep(1:ncol(X))`.
`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.
`nlam`	The number of lambda values. Default is 50.
`intercept`	Should intercept(s) be fitted (default=`TRUE`) or set to zero (`FALSE`)?

lam_min	The smallest value for lambda, as a fraction of lambda.max. Default is 1e-3.
eps	Convergence threshhold. The algorithm iterates until the relative change in any coefficient is less than eps1. Default is 1e-4.
max_step	Maximum number of iterations. Default is 50.
gamma_pen	The tuning parameter of the MCP/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/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.

Value

Bhat	Estimator of coefficients of X.
rss	Residual sum of squares (RSS).
activeX	The active set of X. It is a p dimensional vector.
lambda	The sequence of regularization parameter values in the path.
selectedID	The index of lambda corresponding to lambda_opt.
lambda_opt	The value of lambda with the minimum BIC value.
bic	BIC value used to select variables.
muhat	Estimator of intercept μ. It is NULL if intercept is FALSE.
Chat	Estimator of coefficients of Z. Chat is NULL if Z is NULL.
group	The input group.
Y	Response Y.
X	Design matrix X.

References

A tensor estimation approach to multivariate additive models. Manuscript.

Examples

```
library(tensorMam)

#example 1
n <- 200
q <- 5
s <- 3
p <- 100
B <- matrix(runif(q*s, 2,3), s)
X <- matrix(rnorm(n*p),n,p)
Y <- X[,1:s]%*%B + matrix(rnorm(n*q),n)
fit <- mvrblockwise(Y,X)
fit$activeX
fit$Bhat
which(rowSums(fit$Bhat^2)>0)
```

```
fit$muhat

#example 2
K = 5
n <- 200
q <- 5
s <- 4
p <- 100
B1 <- matrix(runif(q*K, 2,3), K)
B2 <- matrix(0,2*K,q)
B3 <- matrix(runif(q*(s-1)*K, 2,3), (s-1)*K)
B <- rbind(B1,B2,B3)
X <- matrix(rnorm(n*p*K),n)
Y <- X[,1:((s+2)*K)]%*%B + matrix(rnorm(n*q),n)
group <- rep(1:p,each=K)
fit <- mvrblockwise(Y,X,group=group,isPenColumn=TRUE)
which(fit$activeX==1)
fit$Bhat
which(rowSums(fit$Bhat^2)>0)
fit$muhat

#example 3
K = 5
n <- 200
q <- 5
s <- 4
d <- 3
p <- 100
B1 <- matrix(runif(q*K, 2,3), K)
B2 <- matrix(0,2*K,q)
B3 <- matrix(runif(q*(s-1)*K, 2,3), (s-1)*K)
B <- rbind(B1,B2,B3)
C <- matrix(runif(q*d, 1,2), d)
X <- matrix(rnorm(n*p*K),n)
Z <- matrix(rnorm(n*d),n)
Y <- X[,1:((s+2)*K)]%*%B + Z%*%C + matrix(rnorm(n*q),n)
group <- rep(1:p,each=K)
fit <- mvrblockwise(Y,X,Z,group=group,isPenColumn=TRUE)
which(fit$activeX==1)
fit$Bhat
which(rowSums(fit$Bhat^2)>0)
fit$Chat
fit$muhat
```

`mvrcolwise` *Estimate coefficients of high-dimensional multivariate regression for the column-wise*

Description

This function provides the coefficient matrix estimator of high-dimensional multivariate regression (MVR) with penalty `LASSO`, `MCP` or `SCAD`). The tuning parameter is selected by `BIC` (the default), `AIC`, `EBIC`, `CV`, or `GCV`.

Usage

```
mvrcolwise(Y, X ,Z=NULL, criteria="BIC", ncv=10, penalty="LASSO",
            isPenColumn=TRUE, lambda=NULL, nlam=50, intercept=TRUE,
            lam_min=1e-4, eps=1e-6, max_step=50, gamma_pen=2,
            dfmax=NULL, alpha=1)
```

Arguments

Y	The response, a vector of size n or a matrix of size $n \times q$.
X	The covariates to be penalized, a matrix with dimension $n \times p$.
Z	The covariates without penalization, a matrix with dimension $n \times d$. The default is NULL.
criteria	The criteria to be applied to select parameters. Either BIC (the default), AIC, EBIC, CV, or GCV.
ncv	The number of cross-validation folds. Default is 10. If criteria is not CV, ncv is useless.
penalty	The penalty to be applied to the model. Either LASSO (the default), MCP or SCAD.
isPenColumn	A logical value indicating whether the coefficients associating with X_j that affects whole response Y is penalized. Default is TRUE. If isPenColumn is TRUE, the coefficients associating with X_j that affects simultaneously whole response y is penalized for each $j \in \{1, \cdots, p\}$. If isPenColumn is FALSE, the coefficients associating with X_j that affects single response Y_l is penalized for each $j \in \{1, \cdots, p\}$, where $l \in \{1, \cdots, q\}$.
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.
nlam	The number of lambda values. Default is 50.
intercept	Should intercept(s) be fitted (default=TRUE) or set to zero (FALSE)?
lam_min	The smallest value for lambda, as a fraction of lambda.max. Default is 1e-3.
eps	Convergence threshhold. The algorithm iterates until the relative change in any coefficient is less than eps1. Default is 1e-4.
max_step	Maximum number of iterations. Default is 50.
gamma_pen	The tuning parameter of the MCP/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/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.

Value

Bhat	Estimator of coefficients of X.
rss	Residual sum of squares (RSS).
activeX	The active set of X. It is a p dimensional vector.
lambda	The sequence of regularization parameter values in the path.

selectedID	The index of lambda corresponding to lambda_opt.
lambda_opt	The value of lambda with the minimum BIC value.
bic	BIC value used to select variables.
muhat	Estimator of intercept μ. It is NULL if intercept is FALSE.
Chat	Estimator of coefficients of Z. Chat is NULL if Z is NULL.
Y	Response Y.
X	Design matrix X.

References

A tensor estimation approach to multivariate additive models. Manuscript.

Examples

```
library(tensorMam)

#example 1
n <- 200
q <- 5
s <- 3
p <- 100
B <- matrix(runif(q*s, 2,3), s)
X <- matrix(rnorm(n*p),n,p)
Y <- X[,1:s]%*%B + matrix(rnorm(n*q),n)
fit <- mvrcolwise(Y,X)
fit$activeX
fit$Bhat
which(rowSums(fit$Bhat^2)>0)
fit$muhat

#example 2
n <- 200
q <- 5
s <- 3
d <- 3
p <- 100
B <- matrix(runif(q*s, 2,3), s)
C <- matrix(runif(q*d, 1,2), d)
X <- matrix(rnorm(n*p),n,p)
Z <- matrix(rnorm(n*d),n)
Y <- X[,1:s]%*%B + Z%*%C + matrix(rnorm(n*q),n)
fit <- mvrcolwise(Y,X,Z)
fit$activeX
fit$Bhat
which(rowSums(fit$Bhat^2)>0)
fit$Chat
fit$muhat
```

plotfuns — *Plot the estimated curves from* `tensorMam`.

Description

Plot the curves fitted by `mam`, `mam_dr`, `mam_sparse`, and `mam_sparse_dr`

Usage

```
plotfuns(fit,funTrueID,true.curve=FALSE)
```

Arguments

`fit`	Object outputting from `mam`, `mam_dr`, `mam_sparse` or `mam_sparse_dr`.
`funTrueID`	Which function to be plotted. It is a 2-vector. In MAM models, there are $s_0 \times q$ true functions. Thus, the first argument must be smaller than s_0, and the second argument must be smaller than q.
`true.curve`	A Logical flag. Plot both true and estimated curves if true.curve=TRUE. Plot estimated curve only if true.curve=FALSE. Default is FALSE.

Details

This function gives `pq` functional coefficients' estimators of MAM. The singular value matrix of tensor is a $r_1 \times r_2 \times r_3$-tensor. We choose r_1, r_2 and r_3 by `BIC` or `CV`.

References

A tensor estimation approach to multivariate additive models.

See Also

mam, mam_dr, mam_sparse, mam_sparse_dr

Examples

```
n <- 200
p <- 10
q <- 10
s <- 10
K <- 6
s0 <- s
r10=r20=r30=2
S3 <- matrix(runif(r10*r20*r30,3,7),nrow = r30)
T1 <- matrix(rnorm(s0*r10),nrow = s0)
U1 <- qr.Q(qr(T1))
T1 <- matrix(rnorm(K*r20),nrow = K)
U2 <- qr.Q(qr(T1))
T1 <- matrix(rnorm(q*r30),nrow = q)
U3 <- qr.Q(qr(T1))
D3 <- U3%*%S3%*%t(kronecker(U2,U1))
D2 <- TransferModalUnfoldings(D3,3,2,s0,K,q)
mydata <- generateData(n, q, p, s0, D2)
fit <- mam(mydata$Y, mydata$X)
```

```
fit$D2 <- D2
fit$s0 <- s0
fit$X0 <- matrix(runif(100*p),100,p)
plotfuns(fit, c(1,1))
```

TransferModalUnfoldings

Transfer a tensor's modal unfoldings to another.

Description

Transfer a tensor's modal unfoldings to another.

Usage

```
TransferModalUnfoldings(S, d1, d2 , r1, r2, r3)
```

Arguments

S	A mode-d1-unfolding of a tensor with size $r_1 \times r_2 \times r_3$, input unfolding
d1	An integer, the mode of unfolding $S_{(d_1)}$
d2	An integer, the mode of output unfolding $S_{(d_2)}$
r1	The fist dimension of tensor
r2	The second dimension of tensor
r3	The third dimension of tensor

Details

This function transfers an input mode-d1-unfolding $S_{(d_1)}$ to mode-d2-unfolding $S_{(d_2)}$

Value

D	the output mode-d2-unfolding, $S_{(d_2)}$

References

A tensor estimation approach to multivariate additive models.

Examples

```
D1 <- matrix(1:24,nrow = 4) # A tensor unfolding with size 4*6
D2 <- TransferModalUnfoldings(D1,1,2,4,3,2)
```

Index