Package 'comarm'

May 5, 2024

Type Package

Title comarm

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 Multi-omics data integration with multi-view learning via composed tensors. The B-splines are used to approximate component functions. The spline coefficients are rearranged into multiple third-order tensors (MARM) or even a fourth-order tensor. The composed model (COMARM) can be used when the number of covariates in each view is not equal. 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 7.3.1

NeedsCompilation yes

Repository github

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

Encoding UTF-8

R topics documented:

`comarm-package` *Multi-omics data integration with multi-view learning via composed tensors*

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 (MARM) or even a fourth-order tensor (structural MARM). The composed model (COMARM) can 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

Multi-omics data integration with multi-view learning via composed tensors.

`comarm` *Fit composed model (CoMARM) with sparsity assumption and fixed ranks.*

Description

Fits a composed model for multi-view data using B-splines with specified ranks (r_{1g}, r_{2g}, r_{3g} and r_1, r_2, r_3, r_4). This function estimates multiple third-order and a single fourth-order coefficient tensors, applying group sparse penalties such as LASSO, MCP, or SCAD along with a coordinate descent algorithm to optimize the sparsity estimator. The BIC or cross-validation methods are used to select the optimal regularization parameters.

Usage

```
comarm(
  Y,
  X1,
  X2,
  G1 = NULL,
  group = NULL,
  is.fabs = 1,
  K = 6,
  r31 = NULL,
  r32 = NULL,
  r33 = NULL,
  r41 = NULL,
  r42 = NULL,
  r43 = NULL,
  r44 = NULL,
  method = "BIC",
  ncv = 10,
  penalty = "LASSO",
  lambda = NULL,
  D0_t3 = NULL,
  D0_t4 = NULL,
  intercept = TRUE,
  nlam = 50,
  degr = 3,
  lam_min = 0.01,
  eps = 1e-04,
  max_step = 20,
  eps1 = 1e-04,
  max_step1_t3 = 20,
  max_step1_t4 = 20,
  gamma = 2,
  dfmax1 = NULL,
  dfmax2 = NULL,
  alpha = 1,
  vnorm_ratio = 1
)
```

Arguments

Y	A $n \times q$ numeric matrix of responses.
X1	A $n \times p1$ numeric design matrix for the first part of the model.
X2	A $n \times p2$ numeric design matrix for the second part of the model.
G1	Number of views without intergroup correlation.
group	Grouping index for predictors.
is.fabs	Logical indicating if data comes from scenario I.
K	Number of B-spline basis functions.
r31	First dimension rank of third-order tensor.
r32	Second dimension rank of third-order tensor.
r33	Third dimension rank of third-order tensor.
r41	First dimension rank of fourth-order tensor.
r42	Second dimension rank of fourth-order tensor.
r43	Third dimension rank of fourth-order tensor.
r44	Fourth dimension rank of fourth-order tensor.
method	Selection method for model parameters (BIC or CV).
ncv	Number of cross-validation folds.
penalty	Type of penalty (LASSO, MCP, SCAD).
lambda	Sequence of lambda values for regularization.
D0_t3	Initial values for third-order tensor decomposition.
D0_t4	Initial values for fourth-order tensor decomposition.
intercept	Whether to include an intercept.
nlam	Number of lambda values.
degr	Number of knots in the B-spline.
lam_min	Minimum lambda as a fraction of the maximum.
eps	Convergence threshold.
max_step	Maximum number of iterations allowed.
eps1	Convergence threshold for coordinate descent.
max_step1_t3	Maximum iterations for coordinate descent on third-order tensor.
max_step1_t4	Maximum iterations for coordinate descent on fourth-order tensor.
gamma	Tuning parameter for MCP or SCAD.
dfmax1	Maximum number of non-zero coefficients for third-order tensor.
dfmax2	Maximum number of non-zero coefficients for fourth-order tensor.
alpha	Balance parameter between LASSO, MCP/SCAD, and ridge.
vnorm_ratio	Ratio between lambda values for different tensor types.

Value

A list containing model estimates, diagnostic measures, and tensor components.

See Also

comarm.dr

Examples

```
library(comarm)
n <- 200; q <- 5; p <- 100; s1 <- 5; s2 <- 3; G1 <- 1; ng <- 4
group <- rep(1:ng, each = p/ng)
mydata <- comarm.sim.fbs(n, q, p, s1, s2, G1, group, isfixedR = 1)
fit <- with(mydata, comarm(Y, X1, X2, G1, group, is.fabs = 1, K = 6,
              r31 = 2, r32 = 2, r33 = 2, r41 = 2, r42 = 2, r43 = 2, r44 = 2,
              method = "BIC", ncv = 10, penalty = "LASSO", lambda = NULL,
              D0_t3 = NULL, D0_t4 = NULL, intercept = TRUE, nlam = 50, degr = 3,
              lam_min = 0.01, eps = 1e-4, max_step = 20, eps1 = 1e-4,
              max_step1_t3 = 20, max_step1_t4 = 20, gamma = 2, dfmax1 = NULL,
              dfmax2 = NULL, alpha = 1, vnorm_ratio = 1))
```

`comarm.dr` *Fit composed model (CoMARM) 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). This function estimates multiple third-order and fourth-order coefficient tensors. It utilizes group sparse penalties such as LASSO, MCP, or SCAD, and a coordinate descent algorithm to yield a sparsity estimator. The BIC or cross-validation method is applied to optimize regularization parameters, ranks, and the number of B-spline basis functions simultaneously.

Usage

```
comarm.dr(
  Y,
  X1,
  X2,
  G1 = NULL,
  group = NULL,
  is.fabs = 1,
  K_index = NULL,
  r_index = NULL,
  method = "BIC",
  ncv = 10,
  penalty = "LASSO",
  lambda = NULL,
  D0_t3 = NULL,
  D0_t4 = NULL,
  intercept = TRUE,
  nlam = 50,
  degr = 3,
  lam_min = 0.01,
  eps = 1e-04,
  max_step = 20,
  eps1 = 1e-04,
  max_step1_t3 = 20,
  max_step1_t4 = 20,
```

```
  gamma = 2,
  dfmax1 = NULL,
  dfmax2 = NULL,
  alpha = 1,
  vnorm_ratio = 1
)
```

Arguments

Y	A $n \times q$ numeric matrix of responses.
X1	A $n \times p1$ numeric design matrix for the model, corresponding to first set of views.
X2	A $n \times p2$ numeric design matrix for the model, corresponding to second set of views.
G1	The number of views in the first set that we consider without intergroup correlation.
group	A vector of the grouping index for predictors, usually set to `rep(1, p)` by default.
is.fabs	A logical value indicating whether data comes from scenario I.
K_index	User-specified sequence of K, usually `6` for cubic splines.
r_index	A sequence of rank values for the tensors.
method	Method for parameter selection, either `BIC` or `CV`.
ncv	Number of cross-validation folds.
penalty	Type of penalty applied, options: `LASSO`, `MCP`, `SCAD`.
lambda	Sequence of lambda values for regularization.
D0_t3	Initial values for third-order tensor decomposition.
D0_t4	Initial values for fourth-order tensor decomposition.
intercept	Whether to fit an intercept.
nlam	Number of lambda values to use.
degr	Number of knots in B-spline.
lam_min	Minimum lambda as a fraction of the maximum.
eps	Convergence threshold.
max_step	Maximum number of iterations allowed.
eps1	Convergence threshold for coordinate descent.
max_step1_t3	Maximum iterations for coordinate descent on third-order tensor.
max_step1_t4	Maximum iterations for coordinate descent on fourth-order tensor.
gamma	Tuning parameter for MCP/SCAD.
dfmax1	Maximum number of non-zero coefficients for third-order tensor.
dfmax2	Maximum number of non-zero coefficients for fourth-order tensor.
alpha	Balancing parameter between LASSO/MCP/SCAD and ridge.
vnorm_ratio	Ratio between lambda values for different tensor types.

Value

List of model outputs including estimates of tensors, coefficients, and model diagnostics.

See Also

comarm

Examples

```
library(comarm)
n <- 200; q <- 5; p <- 20; s1 <- 5; s2 <- 3; G1 <- 1; ng = 4
group <- rep(1:ng, each = p/ng)
mydata <- comarm.sim.fbs(n, q, p, s1, s2, G1, group)
fit <- with(mydata, comarm.dr(Y = Y, X1 = X1, X2 = X2, G1 = G1,
            group = group, is.fabs = is.fabs, K_index = K,
            r_index = r_index, D0_t3 = D0_t3, D0_t4 = D0_t4, nlam = 5))
```

---

comarm.sim.fbs *Generate scenario I data from composed model (CoMARM).*

Description

This function generates data for scenario I of a composed model (CoMARM), which uses B-splines to model data with known tensor ranks for multiple third-order and fourth-order coefficient tensors. It's suitable for simulations where ranks are fixed and known a priori, allowing detailed study of model behavior under controlled conditions.

Usage

```
comarm.sim.fbs(
  n,
  q,
  p,
  s1,
  s2,
  G1 = NULL,
  group = NULL,
  r310 = 2,
  r320 = 2,
  r330 = 2,
  r410 = 2,
  r420 = 2,
  r430 = 2,
  r440 = 2,
  isfixedR = 0,
  D3 = NULL,
  D44 = NULL,
  K = 6,
  degr = 3,
  sigma2 = NULL,
  seed_id = NULL,
  r1_t3_index = NULL,
  r2_t3_index = NULL,
  r3_t3_index = NULL,
```

```
  r1_t4_index = NULL,
  r2_t4_index = NULL,
  r3_t4_index = NULL,
  r4_t4_index = NULL,
  D0_t3 = NULL,
  D0_t4 = NULL
)
```

Arguments

n	Sample size.
q	Number of responses.
p	Total number of covariates.
s1	Number of true covariates associated with the first set of views.
s2	Number of true covariates associated with the second set of views.
G1	Number of views without intergroup correlations.
group	Grouping index for predictors.
r310	First rank dimension for third-order tensors.
r320	Second rank dimension for third-order tensors.
r330	Third rank dimension for third-order tensors.
r410	First rank dimension for fourth-order tensors.
r420	Second rank dimension for fourth-order tensors.
r430	Third rank dimension for fourth-order tensors.
r440	Fourth rank dimension for fourth-order tensors.
isfixedR	Logical indicating if ranks are fixed.
D3	Initial data for third-order tensor simulation.
D44	Initial data for fourth-order tensor simulation.
K	Number of B-spline basis functions.
degr	Number of knots in B-spline.
sigma2	Error variance.
seed_id	Seed for random number generation.
r1_t3_index	Rank indices for the first dimension of third-order tensors.
r2_t3_index	Rank indices for the second dimension of third-order tensors.
r3_t3_index	Rank indices for the third dimension of third-order tensors.
r1_t4_index	Rank indices for the first dimension of fourth-order tensors.
r2_t4_index	Rank indices for the second dimension of fourth-order tensors.
r3_t4_index	Rank indices for the third dimension of fourth-order tensors.
r4_t4_index	Rank indices for the fourth dimension of fourth-order tensors.
D0_t3	Initial third-order tensor decomposition.
D0_t4	Initial fourth-order tensor decomposition.

Value

A list with the following components:

`Y`	Response matrix, dimensions $n \times q$.
`X1`	First set of design matrices, dimensions $n \times p1$.
`X2`	Second set of design matrices, dimensions $n \times p2$.
`f01`	True function values for the first set of views.
`f02`	True function values for the second set of views.
`group`	Grouping index of predictors.
`D0_t3`	Initialized values for the third-order tensor.
`D0_t4`	Initialized values for the fourth-order tensor.
`...`	Additional algorithm options.

See Also

`comarm.sim.fsin`

Examples

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

`comarm.sim.fsin` *Generate scenario II data from composed model (CoMARM).*

Description

This function generates data for scenario II of a composed model (CoMARM), which is designed to simulate conditions where the true functions are linear combinations of sine and cosine functions. This setup is particularly useful for testing the model's ability to capture complex, periodic relationships within the data.

Usage

```
comarm.sim.fsin(
  n,
  q,
  p,
  s1,
  s2,
  G1 = NULL,
  group = NULL,
  r310 = 2,
  r320 = 2,
  r330 = 2,
  r410 = 2,
```

```
  r420 = 2,
  r430 = 2,
  r440 = 2,
  isfixedR = 0,
  D2 = NULL,
  D42 = NULL,
  K = 6,
  degr = 3,
  sigma2 = NULL,
  seed_id = NULL,
  r1_t3_index = NULL,
  r2_t3_index = NULL,
  r3_t3_index = NULL,
  r1_t4_index = NULL,
  r2_t4_index = NULL,
  r3_t4_index = NULL,
  r4_t4_index = NULL,
  D0_t3 = NULL,
  D0_t4 = NULL
)
```

Arguments

n	Sample size.
q	Number of responses.
p	Total number of covariates.
s1	True covariates for the first set of views.
s2	True covariates for the second set of views.
G1	Number of views without intergroup correlations.
group	Grouping index for predictors.
r310	First dimension rank for third-order tensors.
r320	Second dimension rank for third-order tensors.
r330	Third dimension rank for third-order tensors.
r410	First dimension rank for fourth-order tensors.
r420	Second dimension rank for fourth-order tensors.
r430	Third dimension rank for fourth-order tensors.
r440	Fourth dimension rank for fourth-order tensors.
isfixedR	Logical indicating if ranks are fixed.
D2	Mode of unfolding for the D2 matrix, typically generated randomly.
D42	Mode of unfolding for the D42 matrix, typically generated randomly.
K	Number of B-spline basis functions, typically 6 for cubic splines.
degr	Number of knots in B-spline base functions.
sigma2	Error variance.
seed_id	Seed for random number generation.
r1_t3_index	Rank indices for the first dimension of third-order tensors.
r2_t3_index	Rank indices for the second dimension of third-order tensors.

r3_t3_index	Rank indices for the third dimension of third-order tensors.
r1_t4_index	Rank indices for the first dimension of fourth-order tensors.
r2_t4_index	Rank indices for the second dimension of fourth-order tensors.
r3_t4_index	Rank indices for the third dimension of fourth-order tensors.
r4_t4_index	Rank indices for the fourth dimension of fourth-order tensors.
D0_t3	Initial values for the decomposition of third-order tensors.
D0_t4	Initial values for the decomposition of fourth-order tensors.

Value

A list with components including the response matrix, design matrices for both sets of views, true function matrices, grouping indices, and initial tensor decompositions.

See Also

comarm.sim.fbs

Examples

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

---

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(
  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-04,
  max_step = 20,
  eps1 = 1e-04,
  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 predictors in the model, and the first three predictors are in the same group and the last three predictors are in another one. By default, we set $group = rep(1, p)$.
K	The number of B-spline basis functions, that is the sum of both degrees of basis functions and the number of knots. Default is 6, which means cubic spline.
r1	The first dimension of the singular value matrix of the tensor. Default is 2.
r2	The second dimension of the singular value matrix of the tensor. Default is 2.
r3	The third dimension of the singular 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 matrices 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 threshold. 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 threshold. 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 number of iterates for the 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.

Value

A list containing the following components:

- DEstimator of coefficients $D_{(3)} = (D^1_{(3)}, ..., D^{ng}_{(3)})$ where ng is the number of groups.
- muEstimator of intercept μ.
- S.optA length-ng list including estimator of the core tensor $S_{(3)}$ of each coefficient tensor.
- A.optA length-ng list including estimator of the factor matrix A of each coefficient tensor.
- B.optA length-ng list including estimator of the factor matrix B of each coefficient tensor.
- C.optA length-ng list including estimator of the factor matrix C of each coefficient tensor.
- lambda.seqThe sequence of regularization parameter values in the path.
- lambda_optThe value of lambda with the minimum BIC or CV value.
- rssResidual sum of squares (RSS).
- dfDegrees of freedom.
- activeXThe active set of X. A length-p vector.
- optsOther related parameters used in algorithm. Some of them are set by default.
- opts_penOther related parameters used in algorithm (especially parameters in penalty). Some of them are set by default.

See Also

marm3.dr

Examples

```
library(comarm)
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(
  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-04,
  max_step = 20,
  eps1 = 1e-04,
  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 predictors in the model, and the first three predictors are in the same group and the last three predictors 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`$= 1, \ldots, \min(\lceil \log(n) \rceil, p)$.

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, ..., 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= $1, \ldots, \min(\lceil \log(n) \rceil, q)$.
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.
nlam	The number of lambda values. Default is 50.
degr	The number of knots of B-spline base function. Default is 3.
lam_min	The smallest value for lambda, as a fraction of lambda.max. Default is 0.01.
eps	Convergence threshold. 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 threshold. 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 number of iterates for the 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.

Value

A list containing the following components:

- DEstimator of coefficients $D_{(3)} = (D_{(3)}^1, ..., D_{(3)}^{ng})$.
- muEstimator of intercept μ.
- S.optA length-ng list including estimator of the core tensor $S_{(3)}$ of each coefficient tensor.
- A.optA length-ng list including estimator of the factor matrix A of each coefficient tensor.
- B.optA length-ng list including estimator of the factor matrix B of each coefficient tensor.

- C.optA length-ng list including estimator of the factor matrix C of each coefficient tensor.
- rk_optThe optimal ranks and the number of B-spline basis functions that selected by `BIC`, or `CV`. It is a vector with length 4, which are selected r_1, r_2, r_3, and K.
- lambda.seqThe sequence of regularization parameter values in the path.
- lambda_optThe value of `lambda` with the minimum `BIC` or `CV` value.
- rssResidual sum of squares (RSS).
- dfDegrees of freedom.
- activeXThe active set of X. A length-p vector.
- optsOther related parameters used in algorithm. Some of them are set by default.
- opts_penOther related parameters used in algorithm (especially parameters in penalty). Some of them are set by default.

See Also

`marm3`

Examples

```
library(comarm)
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 using B-splines. The function is designed to simulate multi-view data with specified tensor ranks and a group sparsity structure. The simulation settings allow control over the spline basis functions and the initialization of tensor decompositions.

Usage

```
marm3.sim.fbs(
  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 predictors in the model, and the first three predictors are in the same group and the last three predictors are in another one. By default, set as `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, generated randomly.
K	The number of B-spline basis functions, which is the sum of both the degrees of basis functions and the number of knots. Default is `6`, which indicates cubic splines.
degr	The number of knots of B-spline base function. Default is `3`.
sigma2	Error 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. Default is `r1_index`$= 1, \ldots, \min(\lceil \log(n) \rceil, p)$. Ignored if `isfixedR = 1`.
r2_index	A user-specified sequence of r_2 values. Default is `r2_index = 1, ...,max(K)`. Ignored if `isfixedR = 1`.
r3_index	A user-specified sequence of r_3 values. Default is `r3_index`$= 1, \ldots, \min(\lceil \log(n) \rceil, q)$. Ignored if `isfixedR = 1`.
D0	A user-specified list of initialized values, including ng sub-lists where ng is the number of groups. Each sub-list has four initialized matrices $S_{(3)}$ (called `S`), `A`, `B` and `C`. By default, a list of initialization satisfying fixed ranks is computed randomly.

Value

A list containing:

- YResponse, a $n \times q$-matrix.
- XDesign matrix, a $n \times p$-matrix.
- f0True functions, a $n \times p$-matrix.
- groupThe grouping index of predictors, a length-p vector.
- D0The initialized values.
- ...Other options for the algorithm.

See Also

marm3.sim.fsin

Examples

```
library(comarm)
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. This scenario involves generating data based on sinusoidal transformations of the predictors, incorporating multivariate responses and a structured group sparsity across tensor decompositions.

Usage

```
marm3.sim.fsin(
  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, typically set as group = rep(1, p). Default groups $group = c(1, 1, 1, 2, 2, 2)$, which partitions the predictors into two groups.
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	Logical value indicating if ranks are fixed.
D2	Mode of unfolding $D_{(2)}$, generated randomly by default.
K	The number of B-spline basis functions, combining degrees of basis functions and the number of knots. Default is 6, indicating cubic splines.
degr	The number of knots in the B-spline base function. Default is 3.
sigma2	Error variance. Default is 0.1.
seed_id	Seed for random number generation, affecting reproducibility. Default is 1000.
r1_index	User-specified sequence of r_1 values. Default is r1_index$= 1, \ldots, \min(\lceil \log(n) \rceil, q)$. Ignored if isfixedR = 1.
r2_index	User-specified sequence of r_2 values. Default sequence spans from 1 to the maximum of K. Ignored if isfixedR = 1.
r3_index	User-specified sequence of r_3 values. Default is r3_index$= 1, \ldots, \min(\lceil \log(n) \rceil, q)$. Ignored if isfixedR = 1.
D0	List of initialization values for each group, with each containing initialized matrices $S_{(3)}$, A, B, and C. Randomly computed by default, satisfying fixed ranks.

Value

A list with the following components:

Y	Response matrix, dimensions $n \times q$.
X	Design matrix, dimensions $n \times p$.
f0	Matrix of true functions, dimensions $n \times p$.
group	Vector indicating the grouping index of predictors, length p.
D0	Initialized values used in tensor decomposition.
...	Other algorithmic options, dynamically specified.

See Also

marm3.sim.fbs

Examples

```
library(comarm)
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 integrative 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(
  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-04,
  max_step = 10,
  eps1 = 1e-04,
  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, typically set as `group = rep(1, p)`. Default groups $group = c(1, 1, 1, 2, 2, 2)$, which partitions the predictors into groups.
`K`	The number of B-spline basis functions, typically `6` for cubic splines.
`r1`	The first dimension of the tensor. Default is `2`.

r2	The second dimension of the tensor. Default is 2.
r3	The third dimension of the tensor. Default is 2.
r4	The fourth dimension of the tensor. Default is 2.
method	The method to be applied to select regularization parameters, either BIC or CV. Default is BIC.
ncv	The number of cross-validation folds. Default is 10. If method is not CV, ncv is not used.
penalty	The penalty to be applied to the model, options are LASSO, MCP, or SCAD.
lambda	A sequence of lambda values, calculated by default over a range set by nlam.
D0	Initialization values for the model, includes five matrices: S, A, B, C, and D.
intercept	Indicates if an intercept should be fitted. Default is TRUE.
nlam	The number of lambda values. Default is 20.
degr	The number of knots of the B-spline base function. Default is 3.
lam_min	The smallest lambda value, as a fraction of the largest. Default is 0.01.
eps	Convergence threshold for the overall optimization. Default is 1e-4.
max_step	Maximum number of iterations allowed. Default is 10.
eps1	Convergence threshold for the coordinate descent method. Default is 1e-4.
max_step1	Maximum iterations for the coordinate descent. Default is 10.
gamma	Tuning parameter for the MCP or SCAD penalties.
dfmax	Upper limit on the number of non-zero coefficients.
alpha	Tuning parameter balancing LASSO, MCP/SCAD, and ridge penalties. alpha = 1 favors LASSO/MCP/SCAD, while alpha = 0 (not supported) would indicate pure ridge regression.

Value

A list containing estimates and model parameters, including tensors and matrices of coefficients.

See Also

marm4.dr

Examples

```
library(comarm)
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, r1, r2, r3, r4, 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(
  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-04,
  max_step = 10,
  eps1 = 1e-04,
  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 predictors in the model, and the first three predictors are in the same group and the last three predictors are in another one. By default, $group = rep(1, p)$.
`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$= 1, \ldots, \min(\lceil \log(n) \rceil, p)$.
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, ..., 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$= 1, \ldots, \min(\lceil \log(n) \rceil, ng)$.
r4_index	A user-specified sequence of r_4 values, where r_4 is the fourth dimension of the tensor. Default is r4_index$= 1, \ldots, \min(\lceil \log(n) \rceil, q)$.
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 (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 matrices $S_{(4)}$ (called S), A, B, C, and D. By default, a list of initialization satisfying fixed ranks is computed randomly.
intercept	A logical value indicating whether the intercept is fitted. Default is TRUE.
nlam	The number of lambda values. Default is 20.
degr	The number of knots of B-spline base function. Default is 3.
lam_min	The smallest value for lambda, as a fraction of lambda.max. Default is 0.01.
eps	Convergence threshold. 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 threshold. 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 number of iterations of the coordinate descent method. Default is 20.
gamma	The tuning parameter of the MCP or 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.

Value

A list containing the estimated model parameters and diagnostics, including:

- DEstimator of $D_{(4)}$.
- muEstimator of intercept μ.
- S.optA list including estimator of the core tensor $S_{(4)}$ of each coefficient tensor.

- A.optA list including estimator of the factor matrix A of each coefficient tensor.
- B.optA list including estimator of the factor matrix B of each coefficient tensor.
- C.optA list including estimator of the factor matrix C of each coefficient tensor.
- D.optA list including estimator of the factor matrix D of each coefficient tensor.
- rk_optThe optimal ranks and the number of B-spline basis functions selected by `BIC` or `CV`.
- lambda.seqThe sequence of regularization parameter values in the path.
- lambda_optThe value of `lambda` with the minimum `BIC` or `CV` value.
- rssResidual sum of squares (RSS).
- dfDegrees of freedom.
- activeXThe active set of X. A length-p vector.
- optsOther related parameters used in the algorithm.
- opts_penOther related parameters used in the algorithm, especially parameters in penalty.

See Also

`marm4`

Examples

```
library(comarm)
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 (MARM). This function creates data suitable for analysis with a structural MARM using B-splines with given ranks (r_1, r_2, r_3, r_4) and group sparse penalties such as LASSO, MCP, or SCAD.

Usage

```
marm4.sim.fbs(
  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 vector of the grouping index of predictors, defaults to evenly splitting predictors into two groups.
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 fourth dimension of the tensor. Default is 2.
isfixedR	Indicates if ranks are fixed (TRUE or FALSE).
D44	Mode of unfolding $D_{(4)}$. Generated randomly by default.
K	Number of B-spline basis functions, defaults to 6 for cubic splines.
degr	Number of knots in the B-spline base function. Default is 3.
sigma2	Error variance, default is 0.1.
seed_id	Seed for random number generation, affects data reproducibility. Default is 1000.
r1_index	Sequence of r_1 values, ignored if isfixedR is TRUE.
r2_index	Sequence of r_2 values, ignored if isfixedR is TRUE.
r3_index	Sequence of r_3 values, ignored if isfixedR is TRUE.
r4_index	Sequence of r_4 values, ignored if isfixedR is TRUE.
D0	List of initialization values for matrices $S_{(4)}$, A, B, C, and D.

Value

A list containing generated data elements:

Y	Response matrix, dimensions $n \times q$.
X	Design matrix, dimensions $n \times p$.
f0	Matrix of true functions, dimensions $n \times p$.
group	Vector indicating the grouping index of predictors, length p.
D0	Initialized values used in tensor decomposition.
...	Additional options for the algorithm.

See Also

marm4.sim.fsin

Examples

```
library(comarm)
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 (MARM). This function simulates data based on sinusoidal transformations of predictors, suitable for analysis with a structural MARM model using fixed or varying tensor ranks.

Usage

```
marm4.sim.fsin(
  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 with responses, $s \geq 1$.
group	A vector of grouping index for predictors, splitting predictors into groups. By default, it splits the predictors into two groups.
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 fourth dimension of the tensor. Default is 2.
isfixedR	Indicates if the tensor ranks are fixed (TRUE or FALSE).
D42	Mode of unfolding $D_{(2)}$. Generated randomly by default.
K	Number of B-spline basis functions, typically 6 for cubic splines.
degr	Number of knots in the B-spline base function. Default is 3.
sigma2	Error variance. Default is 0.1.
seed_id	Seed for random number generation, for reproducibility. Default is 1000.
r1_index	User-specified sequence of r_1 values, ignored if isfixedR is TRUE.
r2_index	User-specified sequence of r_2 values, ignored if isfixedR is TRUE.
r3_index	User-specified sequence of r_3 values, ignored if isfixedR is TRUE.
r4_index	User-specified sequence of r_4 values, ignored if isfixedR is TRUE.
D0	List of initialization values for matrices $S_{(4)}$, A, B, C, and D.

Value

A list containing generated data elements:

Y	Response matrix, dimensions $n \times q$.
X	Design matrix, dimensions $n \times p$.
f0	Matrix of true functions, dimensions $n \times p$.
group	Vector indicating the grouping index of predictors, length p.
D0	Initialized values used in tensor decomposition.
...	Additional options for the algorithm.

See Also

marm4.sim.fbs

Examples

```
library(comarm)
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)
```

Index