Package ‘BicMix’

July 9, 2021

Type Package

Title BicMix

Version 0.0-1

Date 2017-10-20

Author Chuan Gao <chuan.gao.cornell@gmail.com>

Maintainer Chuan Gao <chuan.gao.cornell@gmail.com>

Description BicMix is a bayesian statistical method that decomposes a normally distributed matrix into the product of a sparse loading matrix, and a sparse factor matrix. The method employs a three parameter beta distribution that act as a spike and slab prior to induce sparsity to the two matrices. The method further consider the loadings and factors to be drawn from a mixture of sparse and dense component, which allows both the sparse components and the dense hidden confounding to be modeled. The method is especially suitable for deriving context informed functional gene modules, where the context is inferred from the factor matrix, and the sparse gene modules are found on the corresponding loadings. Please refer to our paper for more details (http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1004791).

License GPL (>= 2)

Depends R (>= 2.14)

Collate BicMixWrapper.R

NeedsCompilation yes

RoxygenNote 7.1.1

SystemRequirements GNU make

R topics documented:

BicMixR 2
cal_score_dense 4
cal_score_sparse 5
gen_BicMix_data 5

Index 7

BicMixR — *An algorithm for decomposing a high dimensional matrix into the product of a sparse loading matrix, and a sparse factor matrix.*

Description

An algorithm for decomposing a high dimensional matrix into the product of a sparse loading matrix, and a sparse factor matrix.

Usage

```
BicMixR(
  y = NULL,
  nf = 100,
  a = 0.5,
  b = 0.5,
  c = 0.5,
  d = 0.5,
  e = 0.5,
  f = 0.5,
  itr = 5001,
  rsd = NULL,
  out_itr = 200,
  out_dir = NULL,
  lam_method = "matrix",
  x_method = "dense",
  tol = 0.0000000001,
  qnorm = TRUE,
  nf_min = 5
)
```

Arguments

`y`	matrix to be decomposed, no missing values are allowed, each row is a feature and each column is a sample
`nf`	the number of factors to start with, default to 100
`a`	first shape parameter for the tpb distribution at the local level of the hierarchy, default to 0.5 to recapitulate horseshoe
`b`	second shape parameter for the tpb distribution at the local level of the hierarchy, default to 0.5 to recapitulate horseshoe
`c`	first shape parameter for the tpb distribution at the component specific level of the hierarchy, default to 0.5 to recapitulate horseshoe
`d`	second shape parameter for the tpb distribution at the component specific level of the hierarchy, default to 0.5 to recapitulate horseshoe
`e`	first shape parameter for the tpb distribution at the global level of the hierarchy, default to 0.5 to recapitulate horseshoe
`f`	second shape parameter for the tpb distribution at the global level of the hierarchy, default to 0.5 to recapitulate horseshoe

`itr` the maximum number of iterations the algorithm is allowed to run, default to 5000

`rsd` random seed used for initializing the parameter values, default to a randomly drawn number

`out_itr` iteration number at which the algorithm writes temporary output, default to 200

`out_dir` directory where the algorithm will write temporary output

`lam_method` the method used to update the loading matrix, take values either "matrix" or "element". if "matrix", then all component are updated simultaneously (slower but more stable, don't need as many iterations to converge); if "element", each component is updated sequentially (faster but could be less stable, and need more iterations to converge), default to "matrix"

`x_method` whether induce sparsity on the X matrix, take values either "sparse" or "dense". default to "dense"

`tol` tolerance threshold for convergence, default to 1e-5

`qnorm` whether to quantile-normalize the gene expression matrix, default to TRUE

`nf_min` the minimum number of factors that needed to be kept (when the signals in the data are small, the default shrinkage parameters in BicMix can be too aggressive that zero factors are left. nf_min make sure at lease some factors are kept, default to 5)

Value

lam: the sparse loading matrix

ex: the factor matrix

z: a vector indicating whether the loadings are sparse (1 indicate sparse)

o: a vector indicating whether the factors are sparse (1 indicate sparse)

nf: the number of factors learned by the model

exx: the expected value of the covariance matrix, E(XX^T)

Author(s)

Chuan Gao <chuan.gao.cornell@gmail.com>

References

`http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1004791`

Examples

```
library(BicMix)
## The following is an example on how to use BicMix to obtain biclusters (sparsity is induced on both the loading
## simulate data
data = gen_BicMix_data(std=2, type.factor="mixture",rsd=123)
## Visualize the loading matrix
image(t(data$lam),x=1:ncol(data$lam),y=1:nrow(data$lam),xlab="Loadings",ylab="Samples")
## Visualize the factor matrix
image(t(data$ex),x=1:ncol(data$ex),y=1:nrow(data$ex),xlab="Samples",ylab="Factors")
## run BicMix on the simulated data
dir.create("results")
result = BicMixR(data$y,nf=100,out_dir="results",tol=1e-10,x_method="sparse",rsd=123)
```

```
## calculate a correlation matrix of the estimated loading matrix
## and the true loading matrix. Ideally, there should be one and
## only one big correlation value for a given row and column of the
## correlation matrix
cor.est.real = cor(result$lam[,result$z==1],data$lams)
## visualize the correlation matrix
image(cor.est.real,x=1:nrow(cor.est.real),y=1:ncol(cor.est.real),
xlab="Recovered loadings",ylab="True loadings")
## calculate similarity score of the recovered loading matrix and the true loading matrix
cal_score_sparse(result$lam[,result$z>0.9],data$lams)
## The following is an example on how to use BicMix to obtain clustering of the genes, while keeping the X matrix
## simulate data
data = gen_BicMix_data(std=2, type.factor="dense", rsd = 123)
## perform analysis
result = BicMixR(data$y,nf=100,out_dir="results",tol=1e-10,x_method="dense",rsd=123)
## calculate similarity score of the recovered loading matrix and the true loading matrix
cal_score_sparse(result$lam[,result$z>0.9],data$lams)
```

cal_score_dense *calculate the similarity score of two dense matrices, explained in the SFAmix paper*

Description

calculate the similarity score of two dense matrices, explained in the SFAmix paper

Usage

```
cal_score_dense(m1, m2, precis = FALSE)
```

Arguments

m1 dense matrix 1

m2 dense matrix 2

precis when TRUE, only use the most correlated components in the two matrices to calculate the score, this is to account for the scenario where the two matrices have different dimensions, but all components in the smaller matrix have high correlation with a subset of big matrix

Value

a similarity score

cal_score_sparse *calculate the similarity score of two sparse matrices, explained in the SFAmix paper*

Description

calculate the similarity score of two sparse matrices, explained in the SFAmix paper

Usage

```
cal_score_sparse(m1, m2, precis = FALSE)
```

Arguments

m1	sparse matrix 1
m2	sparse matrix 2
precis	when TRUE, only use the most correlated components in the two matrices to calculate the score, this is to account for the scenario where the two matrices have different dimensions, but all components in the smaller matrix have high correlation with a subset of big matrix

Value

a similarity score

gen_BicMix_data *Simulate a matrix y = lam * ex + err*

Description

Simulate a matrix y = lam * ex + err

Usage

```
gen_BicMix_data(
  std = 2,
  rsd = NULL,
  std.err = 1,
  nf = 15,
  nfs = 10,
  ng = 500,
  ns = 200,
  type.loading = "mixture",
  type.factor = "dense"
)
```

Arguments

`std` standard deviation for the normal distribution of the non-zero entries of the sparse loading

`rsd` random seed

`std.err` standard deviation of the error term

`nf` total number of the loadings

`nfs` number of the sparse loadings

`ng` number of genes

`ns` number of samples

`type.loading` type of loading matrix wanted, "mixture" for a matrix mixed with sparse and dense components, "sparse" for a matrix with only sparse components

`type.factor` type of factor matrix wanted, "mixture" for a matrix mixed with sparse and dense components, "dense" for a matrix with only dense components

Value

a list containing the following

lams: the sparse loadings

lamd: the dense loadings

lam: the loading matrix combining both the sparse and dense loading

ex: the factors matrix combining both the sparse and dense factors

err: matrix of the error term

y: the y matrix calculated as y = lam * ex + err

Index

BicMixR, 2

cal_score_dense, 4
cal_score_sparse, 5

gen_BicMix_data, 5