Package 'pGMGM'

June 28, 2016

Type Package

Title Estimating Multiple Gaussian Graphical Models (GGM) in Penalized Gaussian Mixture Models (GMM)

Version 1.0

Date 2016-06-22

Author Chen Gao, Yunzhang Zhu

Maintainer Chen Gao <gaoxx492@umn.edu>

Description This is an R and C code implementation of the New-SP and New-JGL method of Gao et al. (2016) <DOI:10.1214/16-EJS1135> to perform model-based clustering and multiple graph estimation.

License GPL-2

Depends JGL, mvtnorm, MASS

NeedsCompilation yes

Repository CRAN

Date/Publication 2016-06-28 23:33:21

R topics documented:

`pGMGM-package` *Estimating multiple Gaussian graphical models (GGM) in penalized Gaussian mixture models (GMM).*

Description

This is an R and C code implementation of the New-SP and New-JGL method of Gao et al. (2016) (http://projecteuclid.org/download/pdfview_1/euclid.ejs/1462192266) to perform model-based clustering and multiple graph estimation.

Details

Package:	pGMGM
Type:	Package
Version:	1.0
Date:	2016-06-22
License:	GPL-2

Author(s)

Chen Gao, Yunzhang Zhu

Maintainer: Chen Gao <gaoxx492@umn.edu>

References

Gao, C., hu, Y., Shen, X., and Pan, W. (2016). Estimation of multiple networks in Gaussian mixture models, *Electronic Journal of Statistics*, **10**(1), 1133–1154.
http://projecteuclid.org/download/pdfview_1/euclid.ejs/1462192266
Zhu, Y., Shen, X., and Pan, W. (2014). Structural pursuit over multiple undirected graphs, *Journal of the American Statistical Association*, **109**(508), 1683–1696.
http://www.tandfonline.com/doi/pdf/10.1080/01621459.2014.921182
Danaher, P., Wang, P., and Witten, D. M. (2014). The joint graphical lasso for inverse covariance estimation across multiple classes, *Journal of the Royal Statistical Society, Series B*, **76**(2), 373–397.
http://onlinelibrary.wiley.com/doi/10.1111/rssb.12033/epdf

`jglfit` *Example data set for generating simulated data sets.*

Description

This is the output of applying New-JGL to the gene expression data set used in Gao et al. (2016).

Usage

```
data(jglfit)
```

Format

A data frame with 0 observations on the following 2 variables.

`x` a numeric vector

`y` a numeric vector

Value

A list with the following elements:

`pie`	A vector of the mixing proportion of each component in the model.
`mu`	A matrix that contains the mean of each component in the model in each row.
`covinv`	A list that contains the precision matrices for each component in the model.
`membership`	The class assignment for each observation.
`par`	The optimal value of lambda_1 and lambda_2.

Examples

```
# load data set

data(jglfit)

# mixing proportion
jglfit$pie

# dimension of an estimated precision matrix
dim(jglfit$covinv[[1]])
```

`pGMM` *Estimating multiple Gaussian graphical models (GGM) in penalized Gaussian mixture models (GMM).*

Description

Fit the Gaussian mixture model using either New-SP or New-JGL method. Sparsity and fusion penalty are imposed on the component-specific precision matrices with either non-convex (New-SP) or convex (New-JGL) function. The output contains the class assignment, the mixing proportion, the means and the component-specific precision matrices and the optimal tuning parameter (if a vector of tuning parameter is supplied).

Usage

```
pGMM(Y, k, method, lambda1, lambda2, ncluster, tau=0.01, threshold=1e-5,
  MAX_iter=100, seed=1)
```

Arguments

Y	The input data as an n by p matrix, where n is the number of observations and p is the number of variables.
k	The fold for cross-validation. Default is 5.
method	Either "New-SP" or "New-JGL". "New-SP" uses the non-convex truncated lasso penalty (TLP). "New-JGL" uses the a convex lasso penalty. Default is "New-SP".
lambda1	Tuning parameter for the sparsity penalty, either a single value or a vector. If the input is a vector, cross-validation is applied to select the optimal value.
lambda2	Tuning parameter for the fusion penalty, either a single value or a vector. If the input is a vector, cross-validation is applied to select the optimal value.
ncluster	Number of components in the Gaussian mixture.
tau	The tuning paramter in the non-convex penalty function. Default is 0.01.
threshold	Threshold for convergence. Default is 1e-5.
MAX_iter	Maximum number of iterations. Default is 100.
seed	The seed used when splitting data for cross-validation. Default is 1.

Details

It aims to estimate multiple networks in the presence of sample heterogeneity, where the independent samples (i.e. observations) may come from different and unknown populations or distributions. Specifically, we consider penalized estimation of multiple precision matrices in the framework of a Gaussian mixture model. A major innovation is to take advantage of the commonalities across the multiple precision matrices through possibly nonconvex fusion regularization, which for example makes it possible to achieve simultaneous discovery of unknown disease subtypes and detection of differential gene (dys)regulations in functional genomics. We embed in the EM algorithm one of two recently proposed methods for estimating multiple precision matrices in Gaussian graphical models: one is based on the convex Lasso penalty (Danaher et al., 2014), and the other is based on the con-convex TLP penalty (Zhu et al., 2014); the former is faster, while the latter is less biased and will perform better with a relatively larger sample size.

The majority of the existing literature on mixture modeling focus on regularizing only the mean parameters with diagonal covariance matrices, though some have started considering regularization of the covariance parameters too, all of which, however, do not touch on the key issue of identifying both common and varying substructures of the precision matrices across the components of a mixture model. Since these methods always give different networks for different components unless a common network is assumed, they do not address the question of interest here: which parts of the networks change with the components. Since a fusion penalty is used to shrink multiple networks towards each other, the proposed methods not only are statistically more efficient with information borrowing, but also facilitate interpretation in identifying differential network substructures. In particular, due to the use of a non-convex penalty, the adopted method of Zhu et al. (2014) strives to uncover the commonalities among multiple networks while maintaining their unique substructures too.

Value

pie	A vector of the mixing proportion of each component in the model.
mu	A matrix that contains the mean of each component in the model in each row.
covinv	A list that contains the precision matrices for each component in the model.
membership	The class assignment for each observation.
par	The optimal value of lambda_1 and lambda_2.

Author(s)

Chen Gao, Yunzhang Zhu

References

Gao, C., hu, Y., Shen, X., and Pan, W. (2016). Estimation of multiple networks in Gaussian mixture models, *Electronic Journal of Statistics*, **10**(1), 1133–1154.
http://projecteuclid.org/download/pdfview_1/euclid.ejs/1462192266
Zhu, Y., Shen, X., and Pan, W. (2014). Structural pursuit over multiple undirected graphs, *Journal of the American Statistical Association*, **109**(508), 1683–1696.
http://www.tandfonline.com/doi/pdf/10.1080/01621459.2014.921182
Danaher, P., Wang, P., and Witten, D. M. (2014). The joint graphical lasso for inverse covariance estimation across multiple classes, *Journal of the Royal Statistical Society, Series B*, **76**(2), 373–397.
http://onlinelibrary.wiley.com/doi/10.1111/rssb.12033/epdf

Examples

```
data(jglfit)

# simulate data

set.seed(12345)



Y1 = rmvnorm(1*sum(jglfit$membership==1), jglfit$mu[1,], solve(round(jglfit$covinv[[1]], digits=3)))
Y2 = rmvnorm(1*sum(jglfit$membership==2), jglfit$mu[2,], solve(round(jglfit$covinv[[2]], digits=3)))
Y3 = rmvnorm(1*sum(jglfit$membership==3), jglfit$mu[3,], solve(round(jglfit$covinv[[3]], digits=3)))
Y4 = rmvnorm(1*sum(jglfit$membership==4), jglfit$mu[4,], solve(round(jglfit$covinv[[4]], digits=3)))

Y=rbind(Y1,Y2,Y3,Y4)
p=ncol(Y)
```

```
# test

test = pGMM(Y, k=2, method="New-SP", lambda1=1, lambda2=1, ncluster=4)
```

Index