Package 'multiviewtest'

October 13, 2021

Title Hypothesis Tests for Association Between Subgroups in Two Data Views

Version 2.0.1

Description Tests for association between subgroups in two multivariate data views, two network data views, or a multivariate data view and a network data view. (Reference 1: Gao, L.L., Bien, J., and Witten, D. (2020) ``Are Clusterings of Multiple Data Views Independent?", <doi:10.1093/biostatistics/kxz001> and Reference 2: Gao, L.L., Witten, D., Bien, J. (2021) Testing for Association in Multi-View Network Data, <doi:10.1111/biom.13464>.)

Depends R (>= 3.4)

License GPL-2

Encoding UTF-8

Imports Matrix, stats, mclust, matrixStats, randnet, irlba, doParallel, foreach

RoxygenNote 7.1.1

NeedsCompilation no

Author Lucy L. Gao [aut, cre]

Maintainer Lucy L. Gao <lucy.gao@uwaterloo.ca>

Repository CRAN

Date/Publication 2021-10-13 11:20:02 UTC

R topics documented:

`multiviewtest-package` *multiviewtest: Hypothesis tests for association between subgroups in two data views.*

Description

The multiviewtest package implements three tests for association between subgroups in two data views: 1. The PLRT for association between clusterings of two multivariate data views 2. The P2LRT for association between communities in two network data views 3. The P2LRT for association between communities in a network view, and clusters in a multivariate data view

Details

The PLRT was proposed in Gao, Bien, and Witten (2019) Are Clusterings of Multiple Data Views Independent? Biostatistics, DOI: 10.1093/biostatistics/kxz001.

The P2LRT was proposed in Gao, Witten, Bien (2019+) Testing for Association in Multi-View Network Data, in preparation.

Dependencies

mclust (>= 5.3), matrixStats (>= 0.52.2), Matrix, randnet, doParallel, irbla

multiviewtest functions

`test_indep_clust`: Tests whether clusterings of two multivariate data views are independent. `test_indep_clust_subset`: Tests whether clusterings of two multivariate data views are independent, allowing for clustering on the full data views and comparing on subsets of the data views. `test_indep_com`: Tests whether communities in two network data views are independent. `test_indep_com_clust`: Test whether communities in a network data view and clusters in a multivariate data view are independent. `optimize_over_pi_clust`: An optimization algorithm used in the test of whether clusterings of two data views are independent. `mv_memberships_gen`: Generates subgroup memberships in two data views. `mv_gmm_gen`: Generates two multivariate data views, where each view follows a Gaussian mixture model. `mv_sbm_gen`: Generates two network data views, where each view follows a stochastic block model. `mv_sbm_gmm_gen`: Generates a network view and a multivariate view, where the network view follows a stochastic block model and the multivariate view follows a Gaussian mixture model.

mv_gmm_gen *Generates data from a multi-view Gaussian mixture model*

Description

Generates data from a multi-view Gaussian mixture model with n observations and two views.

Usage

```
mv_gmm_gen(n, Pi, mu1, mu2, Sigma1, Sigma2)
```

Arguments

n	number of observations
Pi	K1 x K2 matrix where the (k, k')th entry contains the probability of an observation belonging to cluster k in View 1 and cluster k' in View 2
mu1	p1 x K1 matrix where the columns contain the K1 cluster means in View 1
mu2	p2 x K2 matrix where the columns contain the K2 cluster means in View 2
Sigma1	p1 x p1 matrix containing the covariance matrix for View 1
Sigma2	p2 x p2 matrix containing the covariance matrix for View 2

Value

A list containing the following components:

data	A list with two items: the view 1 n x p1 multivariate data set and the view 2 n x p2 multivariate data set
clusters	A list with two items: the view 1 cluster memberships and the view 2 cluster memberships

References

Gao, L.L., Bien, J., Witten, D. (2019) Are Clusterings of Multiple Data Views Independent? Biostatistics, <DOI:10.1093/biostatistics/kxz001>

Gao, L.L., Witten, D., Bien, J. Testing for Association in Multi-View Network Data, preprint.

Examples

```
# 25 draws from a two-view Gaussian mixture model where the clusters are independent
n <- 25
Pi <-  tcrossprod(c(0.5, 0.5), c(0.25, 0.25, 0.5))
mu1 <- cbind(c(2, 2), c(-2, 2))
mu2 <- cbind(c(0, 1), c(1, 0), c(-1, 0))
Sigma1 <- diag(rep(1, 2))
Sigma2 <- diag(rep(0.5, 2))

mv_gmm_gen(n, Pi, mu1, mu2, Sigma1, Sigma2)
```

`mv_memberships_gen` *Generates multi-view subgroup memberships*

Description

Generates subgroup membership pairs (Z1, Z2) for a multi-view data set with n observations and two views.

Usage

```
mv_memberships_gen(n, Pi)
```

Arguments

`n`	number of observations
`Pi`	K1 x K2 matrix where the (k, k')th entry contains the probability of an observation belonging to subgroup k in View 1 and subgroup k' in View 2

Value

n x 2 matrix where the first column contains subgroup memberships for View 1 (Z1) and the second column contains subgroup memberships for View 2 (Z2).

References

Gao, L.L., Bien, J., Witten, D. (2019) Are Clusterings of Multiple Data Views Independent? to appear in Biostatistics, <DOI:10.1093/biostatistics/kxz001>

Gao, L.L., Witten, D., Bien, J. Testing for Association in Multi-View Network Data, preprint.

Examples

```
Pi <- tcrossprod(c(0.5, 0.5), c(0.25, 0.25, 0.5))
n <- 25
mv_memberships_gen(n, Pi)
```

`mv_sbm_gen` *Generates data from a stochastic block model for multiple network data views*

Description

Generates data from a stochastic block model for multiple network data views with n observations and two views.

Usage

```
mv_sbm_gen(n, Pi, theta1, theta2, sparse = FALSE)
```

Arguments

n	number of observations
Pi	K1 x K2 matrix where the (k, k')th entry contains the probability of an observation belonging to community k in View 1 and community k' in View 2
theta1	K1 x K1 matrix containing the between-community edge probabilities for View 1
theta2	K2 x K2 matrix containing the between-community edge probabilities for View 2
sparse	If true, return data views in sparseMatrix format

Value

A list containing the following components:

data	A list with two items: the n x n view 1 adjacency matrix and the n x n view 2 adjacency matrix
communities	A list with two items: the view 1 community memberships and the view 2 community memberships

References

Gao, L.L., Witten, D., Bien, J. Testing for Association in Multi-View Network Data, preprint.

Examples

```
# 50 draws from a stochastic block model for two network data views
# where the communities are dependent
n <- 50
Pi <- diag(c(0.5, 0.5))
theta1 <- rbind(c(0.5, 0.1), c(0.1, 0.5))
theta2 <- cbind(c(0.1, 0.5), c(0.5, 0.1))

mv_sbm_gen(n, Pi, theta1, theta2)
```

---

mv_sbm_gmm_gen *Generates data from a stochastic block model for a network view and a multivariate view*

Description

Generates data from a stochastic block model for a network view and a multivariate view with n observations. The data for the multivariate view is drawn from a Gaussian mixture model.

Usage

```
mv_sbm_gmm_gen(n, Pi, theta1, mu2, Sigma2, sparse = FALSE)
```

Arguments

n	number of observations
Pi	K1 x K2 matrix where the (k, k')th entry contains the probability of an observation belonging to community k in View 1 and cluster k' in View 2
theta1	K1 x K1 matrix containing the between-community edge probabilities for View 1
mu2	mu2 p2 x K2 matrix where the columns contain the K2 cluster means in View 2
Sigma2	p2 x p2 matrix containing the covariance matrix for View 2
sparse	If true, return matrix views in sparseMatrix format

Value

A list containing the following components:

data	A list with two items: the view 1 n x n adjacency matrix and the view 2 n x p multivariate data set
communities	A list with two items: the view 1 community memberships and the view 2 cluster memberships

References

Gao, L.L., Witten, D., Bien, J. Testing for Association in Multi-View Network Data, preprint.

Examples

```
# 50 draws from a stochastic block model for a network view and a multivariate view
# where the communities and the clusters are independent
n <- 50
Pi <- tcrossprod(c(0.5, 0.5), c(0.5, 0.5))
theta1 <- rbind(c(0.5, 0.1), c(0.1, 0.5))
mu2 <- cbind(c(2, 2), c(-2, 2))
Sigma2 <- diag(rep(0.5, 2))

mv_sbm_gmm_gen(n, Pi, theta1, mu2, Sigma2)
```

`optimize_over_Pi` *Exponentiated gradient descent for estimating Pi*

Description

Implements the optimization algorithm for solving equation (2.8) in Section 2.3.2 of Gao et al. (2019) "Are Clusterings of Multiple Data Views Independent?" Derivation of the algorithm is given in Appendix B.

Usage

```
optimize_over_Pi(
  logphi1,
  logphi2,
  row,
  col,
  max.iter = 1000,
  stepsz = 0.001,
  Pi.init = NULL
)
```

Arguments

`logphi1`	log(phi1), where the n x K1 matrix phi1 is defined in equation (2.9)
`logphi2`	log(phi1), where the n x K2 matrix phi2 is defined in equation (2.9)
`row`	K1-vector containing the estimated View 1 mixture component probabilities
`col`	K2-vector containing the estimated View 2 mixture component probabilities
`max.iter`	Maximum number of iterations to be run.
`stepsz`	Fixed step size to be used in the optimization; see Appendix B for details.
`Pi.init`	Initializes the optimization algorithm with Pi.init. (Optional)

Value

List of:

`Pi.est`	Estimate of Pi; maximizes the log-likelihood function of X1 and X2.
`obj`	The log-likelihood function evaluated at Pi.est.

References

Gao, L.L., Bien, J., Witten, D. (2019) Are Clusterings of Multiple Data Views Independent? to appear in Biostatistics, <DOI:10.1093/biostatistics/kxz001>

Gao, L.L., Witten, D., Bien, J. Testing for Association in Multi-View Network Data, preprint.

Examples

```
# Generate two-view Gaussian mixture model data with Pi = I/3
set.seed(1)
n <- 100
K <- 3
p <- 2
Sigma <- diag(rep(0.5^2, p))
Pi <- diag(rep(1, K))/K
mu1 <- cbind(c(2, 0), c(0, 2),  c(2, -2))
mu2 <- cbind(c(-2, 0), c(0, -2), c(-2, 2))
dat <- mv_gmm_gen(n, Pi, mu1, mu2, Sigma, Sigma)
view1dat <- dat$data$view1
view2dat <- dat$data$view2
library(mclust)
EM.View1 <- Mclust(view1dat, G=K, modelNames=c("EII"))
EM.View2 <- Mclust(view2dat, G=K, modelNames=c("EII"))
logphi1 <- cdens(modelName="EII", data=view1dat, logarithm=TRUE, parameters=EM.View1$parameters)
logphi2 <- cdens(modelName="EII", data=view2dat, logarithm=TRUE, parameters=EM.View2$parameters)
pi1.est <- EM.View1$parameters$pro
pi2.est <- EM.View2$parameters$pro
Pi.est <- optimize_over_Pi(logphi1, logphi2, pi1.est, pi2.est)
Pi.est$Pi
```

`pl_est_com` *Fits the stochastic block model using maximum pseudolikelihood estimation*

Description

Fits the stochastic block model using maximum pseudolikelihood estimation, as proposed by Amini et. al. (2013). This function implements the conditional pseudolikelihood algorithm from Amini et al. (2013).

Usage

```
pl_est_com(X, K = NULL, max.iter = 1000, tol = 1e-08, parallel = FALSE)
```

Arguments

`X`	n x n adjacency matrix
`K`	number of communities; by default, chosen using the method of Le and Levina (2015)
`max.iter`	maximum number of iterations for the EM algorithm
`tol`	the EM algorithm stops when the relative tolerance is less than this value
`parallel`	An optional argument allowing for parallel computing using the doParallel package

Value

A list containing the following components:

eta	Estimate of eta, a K x K matrix defined in Amini et. al. (2013)
pi	Estimate of the community membership probabilities
ploglik	The maximum of the pseudolikelihood function
logphi	n x K matrix, where (i, k)th entry contains the log p.m.f. of a multinomial random variable with probability vector eta_k (the kth row of eta), evaluated at b_i, which is the ith row of the block compression matrix defined in Amini et. al. (2013)
responsibilities	n x K matrix containing the responsibilities/soft cluster memberships for the n nodes
class	A vector containing n (hard) cluster memberships for the n nodes
converged	whether the algorithm converged to the desired tolerance

References

Amini, A. A., Chen, A., Bickel, P. J., & Levina, E. (2013). Pseudo-likelihood methods for community detection in large sparse networks. The Annals of Statistics, 41(4), 2097-2122.

Le, C. M., & Levina, E. (2015). Estimating the number of communities in networks by spectral methods. arXiv preprint arXiv:1507.00827.

Examples

```
# 50 draws from a stochastic block model for two network data views
# where the communities are dependent
n <- 50
Pi <- diag(c(0.5, 0.5))
theta1 <- rbind(c(0.5, 0.1), c(0.1, 0.5))
theta2 <- cbind(c(0.1, 0.5), c(0.5, 0.1))

dat <- mv_sbm_gen(n, Pi, theta1, theta2)

# Fit SBM to view 1
results <- pl_est_com(X=dat$data$view1, K = 2)
table(results$class, dat$communities$view1)
```

---

test_indep_clust *Pseudo likelihood ratio test for dependent clusterings*

Description

Implements the pseudo likelihood ratio test described in Section 3 of Gao et. al. (2019) "Are Clusterings of Multiple Data Views Independent?" for testing for dependence between clusterings of two data views. Fits Gaussian mixture models in each view.

Usage

```
test_indep_clust(
  x,
  model1 = "EII",
  model2 = "EII",
  K1 = NULL,
  K2 = NULL,
  init1 = NULL,
  init2 = NULL,
  B = 200,
  step = 0.001,
  maxiter = 1000
)
```

Arguments

x	Multi-view data with two views; a list of two numeric vectors (in the case of univariate data) or matrices containing the two data views. In matrix format, rows correspond to observations and columns correspond to variables.
model1	A character string indicating the model to be fitted for Gaussian model-based clustering in view 1 using the function Mclust. The default is "EII" (spherical, equal volume). The help file for mclustModelNames describes the available model options.
model2	A character string indicating indicating the model to be fitted for Gaussian model-based clustering in view 1 using the function Mclust. The default is "EII" (spherical, equal volume). The help file for mclustModelNames describes the available model options.
K1	An optional argument containing the number of clusters in View 1. If left out, then the number of clusters is chosen with BIC as described in Section 2.3.3 of "Are Clusterings of Multiple Data Views Independent?"
K2	An optional argument containing the number of clusters in View 2. If left out, then the number of clusters is chosen with BIC as described in Section 2.3.3 of "Are Clusterings of Multiple Data Views Independent?"
init1	An optional argument containing the model to be fitted in the hierarchical clustering initialization in Gaussian model-based clustering in view 1. The default is "VVV" (ellipsoidal, varying volume, shape, and orientation). The help file for hc describes the available model options.
init2	An optional argument containing the model to be fitted in the hierarchical clustering initialization in Gaussian model-based clustering in view 2. The default is "VVV" (ellipsoidal, varying volume, shape, and orientation). The help file for hc describes the available model options.
B	An integer specifying the number of permutations to use for the permutation procedure. The default number is 200.
step	A numeric value containing the fixed step size to be used in the optimization algorithm for estimating Pi. The default step size is 0.001. See Supplement C of "Are Clusterings of Multiple Data Views Independent?" for details.

maxiter	A numeric value containing the maximum number of iterations to run in the optimization algorithm. The default maximum is 1000.

Value

A list containing the following output components:

K1	The number of clusters in view 1
K2	The number of clusters in view 2
Pi.est	The estimated Pi matrix
PLRstat	The pseudo likelihood ratio test statistic
pval	The p-value
modelfit1	The object of class 'Mclust' corresponding to the model-based clustering fitted in View 1; contains eg. estimated parameters and cluster assignments. The help file for Mclust describes the components of the object.
modelfit2	The object of class 'Mclust' corresponding to the model-based clustering fitted in View 2; contains eg. estimated parameters and cluster assignments. The help file for Mclust describes the components of the object.

References

Fraley C. and Raftery A. E. (2002) Model-based clustering, discriminant analysis and density estimation, Journal of the American Statistical Association, 97/458, pp. 611-631.

Gao, L.L., Bien, J., Witten, D. (2019) Are Clusterings of Multiple Data Views Independent? Biostatistics, DOI: 10.1093/biostatistics/kxz001.

Scrucca L., Fop M., Murphy T. B. and Raftery A. E. (2016) mclust 5: clustering, classification and density estimation using Gaussian finite mixture models, The R Journal, 8/1, pp. 205-233.

Examples

```
set.seed(1)
n <- 50
sig <- 2
p <- 2
K <- 3
mu1 <- cbind(c(2, 0), c(0, 2),  c(2, -2), c(-2, 0), c(0, -2), c(-2, 2))
mu2 <- cbind(c(-2, 0), c(0, -2), c(-2, 2), c(2, 0), c(0, 2), c(2, -2))
# Generates two-view data where the clusters are independent.
x1 <- list(matrix(sig* rnorm(n*p), n, p) + t(mu1)[sample(1:K, n, replace=TRUE), ],
         matrix(sig * rnorm(n*p), n, p) + t(mu2)[sample(1:K, n, replace=TRUE), ])

# Generate two-view data where the clusters are identical.
n <- 70
cl <- sample(1:K, n, replace=TRUE)
x2 <- list(matrix(sig* rnorm(n*p), n, p) + t(mu1)[cl, ],
         matrix(sig * rnorm(n*p), n, p) + t(mu2)[cl, ])

# Run the function on independent data views; we do not reject the null hypothesis.
 # By default, not specifying K1 and K2 means the number of clusters
```

```
 # to use in the test in each view is chosen via BIC.
 # Covariance matrix model specified is shared sigma^2 I covariance matrix in view 1
 # and shared diagonal covariance matrix in view 2.
 # B specifies the number of permutations to do for the permutation test.
 # Covariance matrix model specified for initialization
 # is shared sigma^2 I covariance matrix in view 1
indep1 <- test_indep_clust(x1,model1="EII", model2="EEI",
init1="EII", B=52)
# The estimated cluster parameters in view 1
indep1$modelfit1$parameters
# The cluster assignments in view 2
indep1$modelfit2$classification

# Run the function on identical data views; we reject the null hypothesis
# We specify the number of clusters in each view to use in the test.
# Covariance matrix model specified is shared covariance matrix in view 1
# and shared diagonal covariance matrix in view 2.
# See mclust documentation for more covariance model specification options.
identical2 <- test_indep_clust(x2,model1="EEE", model2="EEI", K1=2, K2=3, B=51)
# P-value
identical2$pval
```

`test_indep_clust_subset` *Pseudo likelihood ratio test for independence of clusterings*

Description

Implements the pseudo likelihood ratio test described in Section 3 of Gao et. al. (2019) "Are Clusterings of Multiple Data Views Independent?" for testing for dependence between clusterings of two data views. Fits Gaussian mixture models in each view, in the case where observations can be removed in one view but not the other (e.g. when view 1 data is available on 1st observation but view 2 data is not available on 2nd observation). Allows for fitting model-based clusterings on the full data views, and comparing these clusterings on subsets of the data views where observations are not removed in both views.

Usage

```
test_indep_clust_subset(
  x,
  model1 = "EII",
  model2 = "EII",
  K1 = NULL,
  K2 = NULL,
  subset1,
  subset2,
  init1 = NULL,
  init2 = NULL,
```

```
  B = 200,
  step = 0.001,
  maxiter = 1000
)
```

Arguments

x	Multi-view data with two views; a list of two numeric vectors (in the case of univariate data) or matrices containing the two data views. In matrix format, rows correspond to observations and columns correspond to variables.
model1	A character string indicating the model to be fitted for Gaussian model-based clustering in view 1 using the function Mclust. The default is "EII" (spherical, equal volume). The help file for mclustModelNames describes the available model options.
model2	A character string indicating indicating the model to be fitted for Gaussian model-based clustering in view 1 using the function Mclust. The default is "EII" (spherical, equal volume). The help file for mclustModelNames describes the available model options.
K1	An optional argument containing the number of clusters in View 1. If left out, then the number of clusters is chosen with BIC as described in Section 2.3.3 of "Are Clusterings of Multiple Data Views Independent?"
K2	An optional argument containing the number of clusters in View 2. If left out, then the number of clusters is chosen with BIC as described in Section 2.3.3 of "Are Clusterings of Multiple Data Views Independent?"
subset1	A numeric vector indicating the rows of the matrix containing the first data view that correspond to observations which have not been removed in the second data view.
subset2	A numeric vector indicating the rows of the matrix containing the second data view that correspond to observations which have not been removed in the first data view.
init1	An optional argument containing the model to be fitted in the hierarchical clustering initialization in Gaussian model-based clustering in view 1. The default is "VVV" (ellipsoidal, varying volume, shape, and orientation). The help file for hc describes the available model options.
init2	An optional argument containing the model to be fitted in the hierarchical clustering initialization in Gaussian model-based clustering in view 2. The default is "VVV" (ellipsoidal, varying volume, shape, and orientation). The help file for hc describes the available model options.
B	An integer specifying the number of permutations to use for the permutation procedure. The default number is 200.
step	A numeric value containing the fixed step size to be used in the optimization algorithm for estimating Pi. The default step size is 0.001. See Appendix B in the Supplementary Materials of "Are Clusterings of Multiple Data Views Independent?" for details.
maxiter	A numeric value containing the maximum number of iterations to run in the optimization algorithm. The default maximum is 1000.

Value

A list containing the following output components:

K1	The number of clusters in view 1
K2	The number of clusters in view 2
Pi.est	The estimated Pi matrix
PLRstat	The pseudo likelihood ratio test statistic
pval	The p-value
modelfit1	The object of class 'Mclust' corresponding to the model-based clustering fitted in View 1; contains eg. estimated parameters and cluster assignments. The help file for Mclust describes the components of the object.
modelfit2	The object of class 'Mclust' corresponding to the model-based clustering fitted in View 2; contains eg. estimated parameters and cluster assignments. The help file for Mclust describes the components of the object.

References

Fraley C. and Raftery A. E. (2002) Model-based clustering, discriminant analysis and density estimation, Journal of the American Statistical Association, 97/458, pp. 611-631.

Gao, L.L., Bien, J., Witten, D. (2019) Are Clusterings of Multiple Data Views Independent? Biostatistics, <DOI:10.1093/biostatistics/kxz001>

Scrucca L., Fop M., Murphy T. B. and Raftery A. E. (2016) mclust 5: clustering, classification and density estimation using Gaussian finite mixture models, The R Journal, 8/1, pp. 205-233.

Examples

```
set.seed(1)
n <- 50
sig <- 2
p <- 2
K <- 3
mu1 <- cbind(c(2, 0), c(0, 2),  c(2, -2), c(-2, 0), c(0, -2), c(-2, 2))
mu2 <- cbind(c(-2, 0), c(0, -2), c(-2, 2), c(2, 0), c(0, 2), c(2, -2))
# Generates two-view data where the clusters are independent.
x1 <- list(matrix(sig* rnorm(n*p), n, p) + t(mu1)[sample(1:K, n, replace=TRUE), ],
        matrix(sig * rnorm(n*p), n, p) + t(mu2)[sample(1:K, n, replace=TRUE), ])
# Generate two-view data where the clusters are identical.
n <- 71
cl <- sample(1:K, n, replace=TRUE)
x2 <- list(matrix(sig* rnorm(n*p), n, p) + t(mu1)[cl, ],
matrix(sig * rnorm(n*p), n, p) + t(mu2)[cl, ])

# Run the function on independent data views; we do not reject the null hypothesis.
 # By default, not specifying K1 and K2 means the number of clusters
 # to use in the test in each view is chosen via BIC.
 # Covariance matrix model specified is shared sigma^2 I covariance matrix in view 1
 # and shared diagonal covariance matrix in view 2.
 # B specifies the number of permutations to do for the permutation test.
```

```
 # Covariance matrix model specified for initialization
 # is shared sigma^2 I covariance matrix in view 1
 # Estimates Gaussian mixture model parameters on x1[[1]] and x1[[2]],
 # and compares the estimated clusterings on the subsetted data
 # x1[[1]][2:48, ] and x1[[1]][2:48, ].
indep1 <- test_indep_clust_subset(x1,model1="EII", model2="EEI",
subset1=2:48, subset2=2:48, init1="EII", B=51)
# The estimated cluster parameters in view 1
indep1$modelfit1$parameters
# The cluster assignments in view 2
indep1$modelfit2$classification
```

`test_indep_com` *Pseudo pseudolikelihood ratio test for dependent communities*

Description

Implements the pseudo pseudolikelihood ratio test described in Section 3 of Gao et. al. (2019) "Testing for Association in Multi-View Network Data" for testing for dependence between communities in two network data views. Fits stochastic block models in each view.

Usage

```
test_indep_com(
  X,
  K1 = NULL,
  K2 = NULL,
  nperm = 200,
  step = 0.001,
  maxiter = 1000,
  parallel = FALSE
)
```

Arguments

`X`	Multi-view data with two views; a list of two n x n adjacency matrices.
`K1`	An optional argument containing the number of communities in View 1. If left out, then the number of communities is chosen with the method of Le and Levina (2015).
`K2`	An optional argument containing the number of communities in View 2. If left out, then the number of communities is chosen with the method of Le and Levina (2015).
`nperm`	An integer specifying the number of permutations to use for the permutation procedure. The default number is 200.
`step`	A numeric value containing the fixed step size to be used in the optimization algorithm for estimating Pi. The default step size is 0.001.

maxiter	A numeric value containing the maximum number of iterations to run in the optimization algorithm. The default maximum is 1000.
parallel	An optional argument; if true, do parallel computing using the doParallel package

Value

A list containing the following output components:

K1	The number of communities in view 1
K2	The number of communities in view 2
Pi.est	The estimated Pi matrix
P2LRstat	The pseudo likelihood ratio test statistic
pval	The p-value
modelfit1	The parameter estimates and community assignment estimates from View 1.
modelfit2	The parameter estimates and community assignment estimates from View 2.

References

Amini, A. A., Chen, A., Bickel, P. J., & Levina, E. (2013). Pseudo-likelihood methods for community detection in large sparse networks. The Annals of Statistics, 41(4), 2097-2122.

Gao, L.L., Witten, D., Bien, J. Testing for Association in Multi-View Network Data, preprint.

Le, C. M., & Levina, E. (2015). Estimating the number of communities in networks by spectral methods. arXiv preprint arXiv:1507.00827.

Examples

```
set.seed(1)
n <- 50
Pi <- diag(c(0.5, 0.5))
theta1 <- rbind(c(0.5, 0.1), c(0.1, 0.5))
theta2 <- cbind(c(0.1, 0.5), c(0.5, 0.1))

# 50 draws from a multi-view SBM with perfectly dependent communities
dat <- mv_sbm_gen(n, Pi, theta1, theta2)

# Test H0: communities are independent
# Data was generated under the alternative hypothesis
results <- test_indep_com(dat$data, nperm=25)
results$pval
```

`test_indep_com_clust` *Pseudo pseudolikelihood ratio test for association between communities in a network view and clusters in a multivariate view*

Description

Implements the pseudo pseudolikelihood ratio test described in Section 4 of Gao et. al. (2019) "Testing for Association in Multi-View Network Data" for testing for dependence between communities in a network data view and cluster in a multivariate view. Fits a stochastic block model in the network view, and a Gaussian mixture model in the multivariate view.

Usage

```
test_indep_com_clust(
  X,
  K1 = NULL,
  K2 = NULL,
  model2 = "EII",
  init2 = NULL,
  nperm = 200,
  step = 0.001,
  maxiter = 1000,
  parallel = FALSE
)
```

Arguments

`X`	Multi-view data with two views; a list of two n x n adjacency matrices.
`K1`	An optional argument containing the number of communities in View 1. If left out, then the number of communities is chosen with the method of Le and Levina (2015).
`K2`	An optional argument containing the number of clusters in View 2. If left out, then the number of clusters is chosen with BIC.
`model2`	A character string indicating the model to be fitted for Gaussian model-based clustering in the multivariate view using the function `Mclust`. The default is `"EII"` (spherical, equal volume). The help file for `mclustModelNames` describes the available model options.
`init2`	An optional argument containing the model to be fitted in the hierarchical clustering initialization in Gaussian model-based clustering in in the multivariate view . The default is `"VVV"` (ellipsoidal, varying volume, shape, and orientation). The help file for `hc` describes the available model options.
`nperm`	An integer specifying the number of permutations to use for the permutation procedure. The default number is 200.
`step`	A numeric value containing the fixed step size to be used in the optimization algorithm for estimating Pi. The default step size is 0.001.

maxiter	A numeric value containing the maximum number of iterations to run in the optimization algorithm. The default maximum is 1000.
parallel	An optional argument allowing for parallel computing using the doParallel package

Value

A list containing the following output components:

K1	The number of communities in view 1
K2	The number of communities in view 2
Pi.est	The estimated Pi matrix
P2LRstat	The pseudo likelihood ratio test statistic
pval	The p-value
modelfit1	The parameter estimates and community assignment estimates from View 1.
modelfit2	The parameter estimates and community assignment estimates from View 2.

References

Amini, A. A., Chen, A., Bickel, P. J., & Levina, E. (2013). Pseudo-likelihood methods for community detection in large sparse networks. The Annals of Statistics, 41(4), 2097-2122.

Fraley C. and Raftery A. E. (2002) Model-based clustering, discriminant analysis and density estimation, Journal of the American Statistical Association, 97/458, pp. 611-631. Gao, L.L., Witten, D., Bien, J. Testing for Association in Multi-View Network Data, preprint.

Le, C. M., & Levina, E. (2015). Estimating the number of communities in networks by spectral methods. arXiv preprint arXiv:1507.00827.

Scrucca L., Fop M., Murphy T. B. and Raftery A. E. (2016) mclust 5: clustering, classification and density estimation using Gaussian finite mixture models, The R Journal, 8/1, pp. 205-233.

Examples

```
# 50 draws from a multi-view SBM, where the clusters
# and the communities are independent
n <- 50
Pi <- tcrossprod(c(0.5, 0.5), c(0.5, 0.5))
theta1 <- rbind(c(0.5, 0.1), c(0.1, 0.5))
mu2 <- cbind(c(2, 2), c(-2, 2))
Sigma2 <- diag(rep(0.5, 2))

dat <- mv_sbm_gmm_gen(n, Pi, theta1, mu2, Sigma2)


# Test H0: communities are independent
# Data was generated under the null hypothesis
results <- test_indep_com_clust(dat$data, nperm=25)
results$pval
```

Index