Package 'IMIX'

October 12, 2022

Type Package

Version 1.1.5

Date 2022-07-13

Title Gaussian Mixture Model for Multi-Omics Data Integration

Description A multivariate Gaussian mixture model framework to integrate multiple types of genomic data and allow modeling of inter-data-type correlations for association analysis. 'IMIX' can be implemented to test whether a disease is associated with genes in multiple genomic data types, such as DNA methylation, copy number variation, gene expression, etc. It can also study the integration of multiple pathways. 'IMIX' uses the summary statistics of association test outputs and conduct integration analysis for two or three types of genomics data. 'IMIX' features statistically-principled model selection, global FDR control and computational efficiency. Details are described in Ziqiao Wang and Peng Wei (2020) <doi:10.1093/bioinformatics/btaa1001>.

License GPL-2

Encoding UTF-8

LazyData true

Depends R (>= 3.5)

Imports crayon,mvtnorm,mixtools,mclust,ggplot2,stats,utils,MASS

URL https://github.com/ziqiaow/IMIX

BugReports https://github.com/ziqiaow/IMIX/issues

RoxygenNote 7.1.1

NeedsCompilation no

Author Ziqiao Wang [aut, cre] (<https://orcid.org/0000-0003-3383-8670>),
Peng Wei [ths] (<https://orcid.org/0000-0001-7758-6116>)

Maintainer Ziqiao Wang <wzqjanet@gmail.com>

Repository CRAN

Date/Publication 2022-07-13 22:10:02 UTC

R topics documented:

`data_p` *P value matrix of two data types*

Description

A dataset with summary statistics p values of 1000 genes for RNAseq and CNV data

Usage

```
data(data_p)
```

Format

A data matrix with 1000 rows and 2 variables:

p.rnaseq P values of data type 1 for all genes

p.cnv P values of data type 2 for all genes

`FDR_control_adaptive` *The Adaptive Procedure for Across-Data-Type FDR Control*

Description

The adaptive procedure for across-data-type FDR control based on the output from IMIX models, this can be directly performed by IMIX function, however, if the user is interested in other mixture models, alpha level or combinations of components, this function would be useful.

Usage

```
FDR_control_adaptive(lfdr, alpha)
```

Arguments

lfdr	Local FDR for each gene of the mixture model results for one component or a combination of components
alpha	Prespecified FDR control level

Value

The estimated mFDR for the target component or component combinaitons and whether the genes is classified in this component/combination after FDR control at alpha level, 1 is yes, 0 is no.

significant_genes_with_FDRcontrol	The output of each gene ordered by the components based on FDR control and within each component ordered by the local FDR, "localFDR" is 1-posterior probability of each gene in the component based on the maximum posterior probability, "class_withoutFDRcontrol" is the classified component based on maximum posterior probability, "class_FDRcontrol" is the classified component based on the across-data-type FDR control at alpha level
estimatedFDR	The estimated marginal FDR value for each component starting from component 2 (component 1 is the global null)
alpha	Prespecified nominal level for the across-data-type FDR control

References

Ziqiao Wang and Peng Wei. 2020. "IMIX: a multivariate mixture model approach to association analysis through multi-omics data integration." Bioinformatics. <doi:10.1093/bioinformatics/btaa1001>.

Examples

```
# First load the data
data("data_p")

# Specify inititial values (this step could be omitted)
mu_input <- c(0,3,0,3)
sigma_input <- rep(1,4)
p_input <- rep(0.5,4)
test1 <- IMIX(data_input = data_p,data_type = "p",mu_ini = mu_input,sigma_ini = sigma_input,
p_ini = p_input,alpha = 0.1,model_selection_method = "AIC")

# Check the selected model based on AIC value
test1$`Selected Model`

# Below is an example for data example 1 in controlling
# the FDR at 0.2 for component 2 & component 4.
# First calculate the local FDR for component 2 & component 4:
lfdr_ge_combined <- 1 - (test1$IMIX_cor_twostep$`posterior prob`[,2] +
test1$IMIX_cor_twostep$`posterior prob`[,4])  # Class 2: (ge+,cnv-); class 4: (ge+,cnv+)
names(lfdr_ge_combined) <- rownames(test1$IMIX_cor_twostep$`posterior prob`)

# Perform the across-data-type FDR control for component 2 & component 4 at alpha level 0.2
```

```
fdr_control1 <- FDR_control_adaptive(lfdr = lfdr_ge_combined, alpha = 0.2)
```

`FDR_control_adaptive_imix` *The Adaptive Procedure for Across-Data-Type FDR Control for IMIX Output*

Description

The adaptive procedure for across-data-type FDR control based on the output from IMIX models, this can be directly performed by IMIX function, however, if the user is interested in other alpha levels, this function would be useful to avoid rerun the IMIX().

Usage

```
FDR_control_adaptive_imix(
  imix_output,
  model = c("IMIX_ind", "IMIX_cor_twostep", "IMIX_cor_restrict", "IMIX_cor"),
  alpha
)
```

Arguments

`imix_output`	The result output from IMIX() function, result controlled at alpha level only for one component each time.
`model`	The target model among "IMIX_ind","IMIX_cor_twostep","IMIX_cor_restrict", and "IMIX_cor". Default is IMIX_ind.
`alpha`	Prespecified FDR control level.

Value

The estimated mFDR for the target component and classify the genes in each component after FDR control at alpha level.

`significant_genes_with_FDRcontrol`
: The output of each gene ordered by the components based on FDR control and within each component ordered by the local FDR, "localFDR" is 1-posterior probability of each gene in the component based on the maximum posterior probability, "class_withoutFDRcontrol" is the classified component based on maximum posterior probability, "class_FDRcontrol" is the classified component based on the across-data-type FDR control at alpha level

`estimatedFDR`
: The estimated marginal FDR value for each component starting from component 2 (component 1 is the global null)

`alpha`
: Prespecified nominal level for the across-data-type FDR control

References

Ziqiao Wang and Peng Wei. 2020. "IMIX: a multivariate mixture model approach to association analysis through multi-omics data integration." Bioinformatics. <doi:10.1093/bioinformatics/btaa1001>.

Examples

```
# First generate the data
library(MASS)
N <- 1000
truelabel <- sample(1:8,prob = rep(0.125, 8),size = N,replace = TRUE)
mu1 <- c(0, 5);mu2 <- c(0, 5);mu3 <- c(0, 5)
mu1_mv <- c(mu1[1], mu2[1], mu3[1]);mu2_mv <- c(mu1[2], mu2[1], mu3[1]);
mu3_mv <- c(mu1[1], mu2[2], mu3[1]);mu4_mv <- c(mu1[1], mu2[1], mu3[2]);
mu5_mv <- c(mu1[2], mu2[2], mu3[1]);mu6_mv <- c(mu1[2], mu2[1], mu3[2])
mu7_mv <- c(mu1[1], mu2[2], mu3[2]);mu8_mv <- c(mu1[2], mu2[2], mu3[2])
cov_sim <- list()
for (i in 1:8) {
  cov_sim[[i]] <- diag(3)
  }
data_z <- array(0, c(N, 3))
data_z[which(truelabel == 1),] <- mvrnorm(n = length(which(truelabel == 1)),
mu = mu1_mv,Sigma = cov_sim[[1]],tol = 1e-6,empirical = FALSE)
data_z[which(truelabel == 2),] <- mvrnorm(n = length(which(truelabel == 2)),
mu = mu2_mv,Sigma = cov_sim[[2]],tol = 1e-6,empirical = FALSE)
data_z[which(truelabel == 3),] <- mvrnorm(n = length(which(truelabel == 3)),
mu = mu3_mv,Sigma = cov_sim[[3]],tol = 1e-6,empirical = FALSE)
data_z[which(truelabel == 4),] <- mvrnorm(n = length(which(truelabel == 4)),
mu = mu4_mv,Sigma = cov_sim[[4]],tol = 1e-6,empirical = FALSE)
data_z[which(truelabel == 5),] <- mvrnorm(n = length(which(truelabel == 5)),
mu = mu5_mv,Sigma = cov_sim[[5]],tol = 1e-6,empirical = FALSE)
data_z[which(truelabel == 6),] <- mvrnorm(n = length(which(truelabel == 6)),
mu = mu6_mv,Sigma = cov_sim[[6]],tol = 1e-6,empirical = FALSE)
data_z[which(truelabel == 7),] <- mvrnorm(n = length(which(truelabel == 7)),
mu = mu7_mv,Sigma = cov_sim[[7]],tol = 1e-6,empirical = FALSE)
data_z[which(truelabel == 8),] <- mvrnorm(n = length(which(truelabel == 8)),
mu = mu8_mv,Sigma = cov_sim[[8]],tol = 1e-6,empirical = FALSE)
rownames(data_z) <- paste0("gene", 1:N)
colnames(data_z) <- c("z.methy", "z.ge", "z.cnv")
dim(data_z)

# Fit the model
test2 <- IMIX(data_input = data_z,data_type = "z",alpha = 0.05,verbose = TRUE)

# Adaptive FDR control at alpha 0.2 for IMIX_cor model
fdr_control2 <- FDR_control_adaptive_imix(imix_output = test2, model = "IMIX_cor",
alpha = 0.2)
```

IMIX *IMIX*

Description

Fitting a multivariate mixture model framework, model selection for the best model, and adaptive procedure for FDR control. Input of summary statistics z scores or p values of two or three data types.

Usage

```
IMIX(
  data_input,
  data_type = c("p", "z"),
  mu_ini = NULL,
  sigma_ini = NULL,
  p_ini = NULL,
  tol = 1e-06,
  maxiter = 1000,
  seed = 10,
  ini.ind = TRUE,
 model = c("all", "IMIX_ind", "IMIX_cor_twostep", "IMIX_cor_restrict", "IMIX_cor"),
  model_selection_method = c("BIC", "AIC"),
  alpha = 0.2,
  verbose = FALSE,
  sort_label = TRUE
)
```

Arguments

data_input	An n x d data frame or matrix of the summary statistics z score or p value, n is the nubmer of genes, d is the number of data types. Each row is a gene, each column is a data type.
data_type	Whether the input data is the p values or z scores, default is p value
mu_ini	Initial values for the mean of the independent mixture model distribution. A vector of length 2*d, d is number of data types. Needs to be in a special format: for example, if d=3, needs to be in the format of (null_1,alternative_1,null_2,alternative_2,null_3,alternative_
sigma_ini	Initial values for the standard deviations of the two components in each data type. A vector of length 2*d, d is number of data types. Needs to be in a special format: for example, if d=3, needs to be in the format of (null_1,alternative_1,null_2,alternative_2,null_3,a
p_ini	Initial values for the proportion of the distribution of the two components in each data type. A vector of length 2*d, d is number of data types. Needs to be in a special format: for example, if d=3, needs to be in the format of (null_1,alternative_1,null_2,alternative_2,null_3,alternative_3).
tol	The convergence criterion. Convergence is declared when the change in the observed data log-likelihood increases by less than epsilon.

`maxiter` The maximum number of iteration, default is 1000

`seed` Set.seed, default is 10

`ini.ind` Use the parameters estimated from IMIX-ind for initial values of other IMIX models, default is TRUE

`model` Which model to use to compute the data, default is all

`model_selection_method`
Model selection information criteria, based on AIC or BIC, default is BIC

`alpha` Prespecified nominal level for global FDR control, default is 0.2

`verbose` Whether to print the full log-likelihood for each iteration, default is FALSE

`sort_label` Whether to sort the component labels in case component labels switched after convergence of the initial values, default is TRUE, notice that if the users chooose not to, they might need to check the interested IMIX model for the converged mean for the true component labels and perform the adaptive FDR control separately for an acurate result

Value

A list of results of IMIX

`IMIX_ind` Results of IMIX_ind, assuming all data types are independent

`IMIX_cor_twostep`
Results of IMIX_cor_twostep, by default the mean is the estimated value of IMIX_ind. If the users are interested to use another mean input, they could directly use function IMIX_cor_twostep and specify the mean

`IMIX_cor` Results of IMIX_cor

`IMIX_cor_restrict`
Results of IMIX_cor_restrict

`AIC/BIC` The AIC and BIC values of all fitted models

`Selected Model` The model with the smallest AIC or BIC value, this is determined by user specifications in the function input "model_selection_method", by default is BIC

`significant_genes_with_FDRcontrol`
The output of each gene ordered by the components based on FDR control and within each component ordered by the local FDR, "localFDR" is 1-posterior probability of each gene in the component based on the maximum posterior probability, "class_withoutFDRcontrol" is the classified component based on maximum posterior probability, "class_FDRcontrol" is the classified component based on the across-data-type FDR control at alpha level

`estimatedFDR` The estimated marginal FDR value for each component starting from component 2 (component 1 is the global null)

`alpha` Prespecified nominal level for the across-data-type FDR control

References

Ziqiao Wang and Peng Wei. 2020. "IMIX: a multivariate mixture model approach to association analysis through multi-omics data integration." Bioinformatics. <doi:10.1093/bioinformatics/btaa1001>.

Tatiana Benaglia, Didier Chauveau, David R. Hunter, and Derek Young. 2009. "mixtools: An R Package for Analyzing Finite Mixture Models." Journal of Statistical Software 32 (6): 1–29. https://www.jstatsoft.org/v32/i06/.

Examples

```
# A toy example
data("data_p")
set.seed(10)
data <- data_p[sample(1:1000,200,replace = FALSE),]
mu_input <- c(0,3,0,3)
sigma_input <- rep(1,4)
p_input <- rep(0.5,4)
test <- IMIX(data_input = data,data_type = "p",mu_ini = mu_input,sigma_ini = sigma_input,
             p_ini = p_input,alpha = 0.1,model_selection_method = "BIC",
             sort_label = FALSE,model = "IMIX_ind")


# The details of this example can be found in Github vignette
# First load the data
data("data_p")

# Specify initial values (this step could be omitted)
mu_input <- c(0,3,0,3)
sigma_input <- rep(1,4)
p_input <- rep(0.5,4)

# Fit IMIX model
test1 <- IMIX(data_input = data_p,data_type = "p",mu_ini = mu_input,sigma_ini = sigma_input,
p_ini = p_input,alpha = 0.1,model_selection_method = "AIC")

#Results
# Print the estimated across-data-type FDR for each component
test1$estimatedFDR

# The AIC and BIC values for each model
test1$`AIC/BIC`

# The best fitted model selected by AIC
test1$`Selected Model`

# The output of IMIX_cor_twostep
str(test1$IMIX_cor_twostep)

# The output of genes with local FDR values and classified components
dim(test1$significant_genes_with_FDRcontrol)
head(test1$significant_genes_with_FDRcontrol)
```

IMIX_cor *IMIX-Cor*

Description

Fitting a correlated multivariate mixture model. Input of summary statistics z scores or p values of two or three data types.

Usage

```
IMIX_cor(
  data_input,
  data_type = c("p", "z"),
  g = 8,
  mu_vec,
  cov,
  p,
  tol = 1e-06,
  maxiter = 1000,
  seed = 10,
  verbose = FALSE
)
```

Arguments

data_input	An n x d data frame or matrix of the summary statistics z score or p value, n is the nubmer of genes, d is the number of data types. Each row is a gene, each column is a data type.
data_type	Whether the input data is the p values or z scores, default is p value
g	The number of components, default is 8 for three data types
mu_vec	A list of initial values for the mean vectors for each component. If there are three data types and 8 components, then the initial is a list of 8 mean vectors, each vector is of length 3.
cov	A list of initial values for the covariance matrices. If there are three data types and 8 components, then the initial is a list of 8 covariance matrices, each matix is 3*3.
p	Initial value for the proportion of the distribution in the Gaussian mixture model
tol	The convergence criterion. Convergence is declared when the change in the observed data log-likelihood increases by less than epsilon.
maxiter	The maximum number of iteration, default is 1000
seed	set.seed, default is 10
verbose	Whether to print the full log-likelihood for each iteration, default is FALSE

Value

A list of the results of IMIX-cor

`posterior prob`	Posterior probability matrix of each gene for each component
`Full LogLik all`	Full log-likelihood of each iteration
`Full MaxLogLik final`	The final log-likelihood of the converged model
`iterations`	Number of iterations run
`pi`	Estimated proportion of each component, sum to 1
`mu`	A list of estimated mean vectors of each component for each data type, each list corresponds to one component
`cov`	A list of estimated variance-covariance matrix of each component
`g`	Number of components

References

Ziqiao Wang and Peng Wei. 2020. "IMIX: a multivariate mixture model approach to association analysis through multi-omics data integration." Bioinformatics. <doi:10.1093/bioinformatics/btaa1001>.

`IMIX_cor_restrict` *IMIX-Cor-Restrict*

Description

Fitting a correlated multivariate mixture model with restrictions on the mean. Input of summary statistics z scores or p values of two or three data types.

Usage

```
IMIX_cor_restrict(
  data_input,
  data_type = c("p", "z"),
  mu,
  cov,
  p,
  tol = 1e-06,
  maxiter = 1000,
  seed = 10,
  verbose = FALSE
)
```

Arguments

data_input	An n x d data frame or matrix of the summary statistics z score or p value, n is the nubmer of genes, d is the number of data types. Each row is a gene, each column is a data type.
data_type	Whether the input data is the p values or z scores, default is p value
mu	Initial value for the mean of the independent mixture model distribution. A vector of length 2*d, d is number of data types. Needs to be in a special format that corresponds to the initial value of mu, for example, if d=3, needs to be in the format of (null_1,alternative_1,null_2,alternative_2,null_3,alternative_3).
cov	A list of initial values for the covariance matrices. If there are three data types and 8 components, then the initial is a list of 8 covariance matrices, each matix is 3*3.
p	Initial value for the proportion of the distribution in the Gaussian mixture model
tol	The convergence criterion. Convergence is declared when the change in the observed data log-likelihood increases by less than epsilon.
maxiter	The maximum number of iteration, default is 1000
seed	set.seed, default is 10
verbose	Whether to print the full log-likelihood for each iteration, default is FALSE

Value

A list of the results of IMIX-cor-restrict

posterior prob	Posterior probability of each gene for each component
Full LogLik all	Full log-likelihood of each iteration
Full MaxLogLik final	The final log-likelihood of the converged model
iterations	Number of iterations run
pi	Estimated proportion of each component, sum to 1
mu	Estimated mean for the null and alternative of each data type: for two data types (mu10,mu11,mu20,mu21), three data types (mu10,mu11,mu20,mu21,mu30,mu31), mui0 is the null for data type i, mui1 is the alternative for data type i.
cov	A list of estimated variance-covariance matrix of each component

References

Ziqiao Wang and Peng Wei. 2020. "IMIX: a multivariate mixture model approach to association analysis through multi-omics data integration." Bioinformatics. <doi:10.1093/bioinformatics/btaa1001>.

IMIX_cor_twostep *IMIX-Cor-Twostep*

Description

Fitting a correlated multivariate mixture model with fixed mean from estimated parameters of IMIX-ind. Input of summary statistics z scores or p values of two or three data types.

Usage

```
IMIX_cor_twostep(
  data_input,
  data_type = c("p", "z"),
  g = 8,
  mu_vec,
  cov,
  p,
  tol = 1e-06,
  maxiter = 1000,
  seed = 10,
  verbose = FALSE
)
```

Arguments

data_input	An n x d data frame or matrix of the summary statistics z score or p value, n is the nubmer of genes, d is the number of data types. Each row is a gene, each column is a data type.
data_type	Whether the input data is the p values or z scores, default is p value
g	The number of components, default is 8 for three data types
mu_vec	Input of the mean value output from IMIX-Ind result, a list of the mean vectors for each component.
cov	A list of initial values for the covariance matrices. If there are three data types and 8 components, then the initial is a list of 8 covariance matrices, each matix is 3*3.
p	Initial value for the proportion of the distribution in the Gaussian mixture model
tol	The convergence criterion. Convergence is declared when the change in the observed data log-likelihood increases by less than epsilon.
maxiter	The maximum number of iteration, default is 1000
seed	set.seed, default is 10
verbose	Whether to print the full log-likelihood for each iteration, default is FALSE

Value

A list of the results of IMIX-cor-twostep

`posterior prob`	Posterior probability matrix of each gene for each component
`Full LogLik all`	Full log-likelihood of each iteration
`Full MaxLogLik final`	The final log-likelihood of the converged model
`iterations`	Number of iterations run
`pi`	Estimated proportion of each component, sum to 1
`mu`	A list of mean vectors of each component for each data type, this is the prespecified mean
`cov`	A list of estimated variance-covariance matrix of each component
`g`	Number of components

References

Ziqiao Wang and Peng Wei. 2020. "IMIX: a multivariate mixture model approach to association analysis through multi-omics data integration." Bioinformatics. <doi:10.1093/bioinformatics/btaa1001>.

`IMIX_ind` *IMIX-ind*

Description

Fitting an independent mixture model with restrictions on mean and variance. Input of summary statistics z scores or p values of two or three data types.

Usage

```
IMIX_ind(
  data_input,
  data_type = c("p", "z"),
  mu,
  sigma,
  p,
  tol = 1e-06,
  maxiter = 1000,
  seed = 10,
  verbose = FALSE
)
```

Arguments

data_input	An n x d data frame or matrix of the summary statistics z score or p value, n is the nubmer of genes, d is the number of data types. Each row is a gene, each column is a data type.
data_type	Whether the input data is the p values or z scores, default is p value
mu	Initial value for the mean of each component of the independent mixture model distribution. A vector of length 2*d, d is number of data types. Needs to be in a special format that corresponds to the initial value of mu, for example, if d=3, needs to be in the format of (null_1,alternative_1,null_2,alternative_2,null_3,alternative_3).
sigma	Initial value for the standard deviation of each component of the independent mixture model distribution. A vector of length 2*d, d is number of data types. Needs to be in a special format that corresponds to the initial value of mu, for example, if d=3, needs to be in the format of (null_1,alternative_1,null_2,alternative_2,null_3,alternative_3).
p	Initial value for the proportion of the distribution in the Gaussian mixture model
tol	The convergence criterion. Convergence is declared when the change in the observed data log-likelihood increases by less than epsilon.
maxiter	The maximum number of iteration, default is 1000
seed	set.seed, default is 10
verbose	Whether to print the full log-likelihood for each iteration, default is FALSE

Value

A list of the results of IMIX-ind

posterior prob	Posterior probability matrix of each gene for each component
Full LogLik all	Full log-likelihood of each iteration
Full MaxLogLik final	The final log-likelihood of the converged model
iterations	Number of iterations run
pi	Estimated proportion of each component, sum to 1
mu	Estimated mean for the null and alternative of each data type: for two data types (mu10,mu11,mu20,mu21), three data types (mu10,mu11,mu20,mu21,mu30,mu31), mui0 is the null for data type i, mui1 is the alternative for data type i.
sigma	Estimated standard deviation for the null and alternative of each data type: for two data types (sigma10,sigma11,sigma20,sigma21), three data types (sigma10,sigma11,sigma20,sigma2 sigmai0 is the null for data type i, sigmai1 is the alternative for data type i.

References

Ziqiao Wang and Peng Wei. 2020. "IMIX: a multivariate mixture model approach to association analysis through multi-omics data integration." Bioinformatics. <doi:10.1093/bioinformatics/btaa1001>.

`model_selection` *Model Selection*

Description

Model selection for sub-model outputs in IMIX, this step is to calculate the AIC or BIC values for one model

Usage

```
model_selection(
  loglik,
  n,
  g = 4,
  d = 2,
 modelname = c("IMIX_ind", "IMIX_ind_unrestrict", "IMIX_cor_twostep", "IMIX_cor",
    "IMIX_cor_restrict")
)
```

Arguments

`loglik`	Full log likelihood, result output from IMIX or a sub model in IMIX: 'Full MaxLogLik final'
`n`	Total number of genes
`g`	Number of components
`d`	Number of data types
`modelname`	The model name, default is IMIX_ind

Value

AIC/BIC values of the target model

References

Ziqiao Wang and Peng Wei. 2020. "IMIX: a multivariate mixture model approach to association analysis through multi-omics data integration." Bioinformatics. <doi:10.1093/bioinformatics/btaa1001>.

Examples

```
# First load the data
data("data_p")

# Specify the initial values
mu_input <- c(0,3,0,3)
sigma_input <- rep(1,4)
p_input <- rep(0.5,4)
```

```
# Fit the IMIX model
test1 <- IMIX(data_input = data_p,data_type = "p",mu_ini = mu_input,sigma_ini = sigma_input,
p_ini = p_input,alpha = 0.1,model_selection_method = "AIC")

# Calculate the AIC and BIC values for IMIX_ind with two data types and four components
model_selection(test1$IMIX_ind$`Full MaxLogLik final`,
n=dim(test1$IMIX_ind$`posterior prob`)[1],g=4,d=2, "IMIX_ind")
```

`model_selection_component` *Component Selection*

Description

Model selection for components based on AIC and BIC values for models in IMIX

Usage

```
model_selection_component(
  data_input,
  data_type = c("p", "z"),
  tol = 1e-06,
  maxiter = 1000,
  seed = 10,
  verbose = FALSE
)
```

Arguments

`data_input`	An n x d data frame or matrix of the summary statistics z score or p value, n is the nubmer of genes, d is the number of data types. Each row is a gene, each column is a data type.
`data_type`	Whether the input data is the p values or z scores, default is p value
`tol`	The convergence criterion. Convergence is declared when the change in the observed data log-likelihood increases by less than epsilon.
`maxiter`	The maximum number of iteration, default is 1000
`seed`	set.seed, default is 10
`verbose`	Whether to print the full log-likelihood for each iteration, default is FALSE

Value

Selected number of components based on AIC and BIC

`Component_Selected_AIC`
: Selected number of components by AIC with the smallest AIC value among all components and models

`Component_Selected_BIC`
: Selected number of components by BIC with the smallest BIC value among all components and models

`AIC/BIC`
: The AIC and BIC values for all components for IMIX_ind_unrestrict, IMIX_cor_twostep, and IMIX_cor

`IMIX_ind_unrestrict`
: A list of the IMIX_ind_unrestrict for all components 1,2,...2^d, this step was fitted using R package "Mclust", more details of the output can be found there

`IMIX_cor_twostep`
: A list of the IMIX_cor_twostep for all components 1,2,...2^d, here, the mean is the estimated value of IMIX_ind_unrestrict

`IMIX_cor`
: A list of the IMIX_cor_twostep for all components 1,2,...2^d

References

Ziqiao Wang and Peng Wei. 2020. "IMIX: a multivariate mixture model approach to association analysis through multi-omics data integration." Bioinformatics. <doi:10.1093/bioinformatics/btaa1001>.
Luca Scrucca, Michael Fop, T. Brendan Murphy, and Adrian E. Raftery. 2016. "mclust 5: Clustering, Classification and Density Estimation Using Gaussian Finite Mixture Models." The R Journal 8 (1): 289–317. <doi:10.32614/RJ-2016-021>.

Examples

```
# A toy example
data("data_p")
set.seed(10)
data <- data_p[sample(1:1000,20,replace = FALSE),]
select_comp0 <- model_selection_component(data, data_type = "p", seed = 20)


# First load the data
data("data_p")

# Perform model selections on the data
select_comp1 = model_selection_component(data_p, data_type = "p", seed = 20)
```

---

plot_component *Plot the AIC or BIC Values for Model Selection*

Description

Plot the result output of model selection for components based on AIC and BIC values in IMIX

Usage

```
plot_component(res_select, type = c("AIC", "BIC"))
```

Arguments

res_select	Result output from function model_selection_component()
type	Which information criteria to use for plot

Value

Plot for the model selection of components

References

Ziqiao Wang and Peng Wei. 2020. "IMIX: a multivariate mixture model approach to association analysis through multi-omics data integration." Bioinformatics. <doi:10.1093/bioinformatics/btaa1001>.

Examples

```
# First load the data
data("data_p")

# Perform model selections on the data
select_comp1 <- model_selection_component(data_p, data_type = "p", seed = 20)

# Make a plot for BIC values
plot_component(select_comp1, type = "BIC")
```

Index