

# Package: cobiclust (via r-universe)

August 20, 2024

**Type** Package

**Title** Biclustering via Latent Block Model Adapted to Overdispersed Count Data

**Version** 0.1.0

**Description** Implementation of a probabilistic method for biclustering adapted to overdispersed count data. It is a Gamma-Poisson Latent Block Model. It also implements two selection criteria in order to select the number of biclusters.

**License** GPL-3

**URL** <https://github.com/julieaubert/cobiclust>

**BugReports** <https://github.com/julieaubert/cobiclust/issues>

**Imports** cluster, stats

**Suggests** spelling, testthat

**Encoding** UTF-8

**Language** en-US

**LazyData** true

**RoxygenNote** 7.1.1

**Repository** <https://julieaubert.r-universe.dev>

**RemoteUrl** <https://github.com/julieaubert/cobiclust>

**RemoteRef** HEAD

**RemoteSha** 834fdb2a4e429a27315b4f3e3f87b0898609408a

## Contents

cobiclust . . . . .	2
selection_criteria . . . . .	3

<b>Index</b>	<b>4</b>
--------------	----------

---

`cobiclust`*Perform a biclustering adapted to overdispersed count data.*

---

**Description**

Perform a biclustering adapted to overdispersed count data.

**Usage**

```
cobiclust(  
  x,  
  K = 2,  
  G = 3,  
  nu_j = NULL,  
  a = NULL,  
  akg = FALSE,  
  cvg_lim = 1e-05,  
  nbiter = 5000,  
  tol = 1e-04  
)
```

**Arguments**

<code>x</code>	the input matrix of observed data.
<code>K</code>	an integer specifying the number of groups in rows.
<code>G</code>	an integer specifying the number of groups in columns.
<code>nu_j</code>	a vector of . The length is equal to the number of columns.
<code>a</code>	an numeric.
<code>akg</code>	a logical variable indicating whether to use a common dispersion parameter ( <code>akg = FALSE</code> ) or a dispersion parameter per cocluster ( <code>akg = TRUE</code> ).
<code>cvg_lim</code>	a number specifying the threshold used for convergence criterion ( <code>cvg_lim = 1e-05</code> by default).
<code>nbiter</code>	the maximal number of iterations for the global loop of variational EM algorithm ( <code>nbiter = 5000</code> by default).
<code>tol</code>	the level of relative iteration convergence tolerance ( <code>tol = 1e-04</code> by default).

**Value**

An object of class `cobiclustering`

**See Also**

[cobiclustering](#) for the `cobiclustering` class.

**Examples**

```

npc <- c(50, 40) # nodes per class
KG <- c(2, 3) # classes
nm <- npc * KG # nodes
Z <- diag( KG[1]) %% matrix(1, npc[1], 1)
W <- diag(KG[2]) %% matrix(1, npc[2], 1)
L <- 70 * matrix( runif( KG[1] * KG[2]), KG[1], KG[2])
M_in_expectation <- Z %% L %% t(W)
size <- 50
M<-matrix(
  rnbinom(
    n = length(as.vector(M_in_expectation)),
    mu = as.vector(M_in_expectation), size = size)
  , nm[1], nm[2])
rownames(M) <- paste("OTU", 1:nrow(M), sep = "_")
colnames(M) <- paste("S", 1:ncol(M), sep = "_")
res <- cobiclust(M, K = 2, G = 3, nu_j = rep(1,120), a = 1/size, cvg_lim = 1e-5)

```

---

selection\_criteria      *Calculate selection criteria.*

---

**Description**

Calculate selection criteria.

**Usage**

```
selection_criteria(x, K, G)
```

**Arguments**

x	The output of the cobiclust function.
K	The number of groups in rows.
G	The number of groups in columns.

**Value**

A dataframe with 7 columns.

vICL the vICL selection criterion.

BIC the BIC selection criterion.

penKG the value of the BIC penalty.

lb the value of the lower bound of the log-likelihood.

entZW the value of the entropy of the latent variables Z and W.

K the number of groups in rows.

G the number of groups in columns.

# Index

cobiclust, 2

cobiclustering, 2

selection\_criteria, 3