

# Package ‘ILSM’

July 30, 2025

**Title** Analyze Interconnection Structure of Tripartite Interaction Networks

**Version** 1.1.0.1

**Description** In view of the analysis of the structural characteristics of the tripartite network has been complete, however, there is still a lack of a unified operation that can quickly obtain the corresponding characteristics of the tripartite network.

To solve this insufficiency, 'ILSM' was designed for supporting calculating such metrics of tripartite networks by functions of this R package.

**License** MIT + file LICENSE

**URL** <https://github.com/WeichengSun/ILSM>

**BugReports** <https://github.com/WeichengSun/ILSM/issues>

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**Depends** R (>= 3.5.0)

**Imports** stats, vegan, igraph, plot.matrix

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0)

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**NeedsCompilation** no

**Author** Weicheng Sun [aut, cre],  
Chuan Yan [aut],  
Yangyang Zhao [aut]

**Maintainer** Weicheng Sun <sunwch2023@lzu.edu.cn>

**Repository** CRAN

**Date/Publication** 2025-07-30 16:40:02 UTC

## Contents

build_toy_net	2
coid	3
cois	5
hc	7
icmotif_count	8
icmotif_role	10
ig_ddom	12
ig_icmotif_count	14
ig_icmotif_role	16
ig_overlap_guild	18
Multi_motif	20
node_icc	21
pc	23
poc	25
PPH_Coltparkmeadow	26
role_sim	27
trigraph_from_mat	28
tri_null	29

<b>Index</b>	<b>31</b>
--------------	-----------

---

build_toy_net	<i>Generating an example tripartite network randomly</i>
---------------	--

---

### Description

Generate a binary tripartite network with three groups of nodes (a-, b- and c-nodes) and two subnetworks (P and Q). Subnetwork P contains links between a- and b-nodes; Subnetwork Q contains links between b- and c-nodes; b-nodes (some of which are connector nodes) are shared nodes between two subnetworks.

### Usage

```
build_toy_net(N_a, N_b, N_c, Co, output_matrices = FALSE)
```

### Arguments

N_a	The number of nodes in the a-node group.
N_b	The number of nodes in the b-node group.
N_c	The number of nodes in the c-node group.
Co	The probability of creating a link between any two nodes. It ranges from 0 to 1.
output_matrices	Logical. Whether to output the entire adjacency matrix of the network and subnetworks. Defaults to FALSE.

**Value**

Return a random binary tripartite network.

**References**

Pilosof, S., Porter, M., Pascual, M. et al. The multilayer nature of ecological networks. *Nat Ecol Evol* 1, 0101 (2017). <https://doi.org/10.1038/s41559-017-0101>

**Examples**

```
set.seed(12)
Net <- build_toy_net(11,15,16,0.2)
plot(Net)
```

```
set.seed(12)
Net <- build_toy_net(11,15,16,0.2,output_matrices=TRUE)
Net
```

---

coid

*Correlation of degree of connector nodes*


---

**Description**

Calculating correlation of degree of connector nodes.

**Usage**

```
coid(
  network.or.subnet_mat1,
  subnet_mat2 = NULL,
  weighted = FALSE,
  weight_type = "shannon",
  method = "kendall"
)
```

**Arguments**

network.or.subnet_mat1	An igraph object or matrix. An "igraph" object with node attribute 'level' or a matrix representing one subnetwork. See details.
subnet_mat2	A matrix representing one subnetwork.
weighted	Logical. Default to FALSE. If TRUE, the degree of a connector node is replaced with Shannon diversity or sum of its link strengths.
weight_type	For weighted=TRUE only, supporting "shannon" or "sum".
method	Correlation method ("pearson", "kendall" or "spearman"). Default to "kendall".

## Details

In this package, a tripartite network contains three groups of nodes (a-nodes,b-nodes,c-nodes) and two subnetworks (P includes the links between a-nodes and b-nodes, Q includes the links between b-nodes and c-nodes). Connector nodes belong to b-nodes.

This function follows Sauve et al.(2016) to calculate the correlation of interaction degree (or weighted degree ) of connector nodes. For the binary network, connector nodes' degree is calculated in each subnetwork. For the quantitative network, Shannon diversity or sum of link strength for each connector node is calculated. Three correlation methods are supported. Kendall correlation is recommended following Sauve et al.(2016).

Two types of inputs `network`. or `.subnet_mat1` can be processed:

- An "igraph" object with node attribute 'level' (0 for a-nodes, 1 for b-nodes,2 for c-nodes). If the input is a weighted network, the edge should have a 'weight' attribute.
- Or a matrix representing subnetwork P, and must be input with `subnet_mat2` representing subnetwork Q.

If the inputs are two matrices, please make sure the rows of `network`. or `.subnet_mat1` and `subnet_mat2` both represent the groups of connector species,i.e, the b-group species. If both matrices have row names, then the function matches row names to produce connector nodes. Otherwise, row numbers are assigned to row names and matched. Within the two matrices (P and Q), columns represents a-group nodes and c-group nodes respectively. Elements in matrices are non-zero values if two nodes are linked with or without weights, and 0 otherwise.

## Value

Return a numeric value representing correlation of interaction degree for connector nodes.

## References

Sauve, A. M., Thebault, E., Pocock, M. J., & Fontaine, C. (2016). How plants connect pollination and herbivory networks and their contribution to community stability. *Ecology*, 97(4), 908-917.

## Examples

```
## generate a random binary tripartite network
set.seed(12)
Net <- build_toy_net(11,15,16,0.2)
coid(Net)

## empirical network
data(PPH_Coltparkmeadow)
Net <- PPH_Coltparkmeadow
coid(Net)
set.seed(13)
library(igraph)
E(Net)$weight<-runif(length(E(Net)),0.1,1)#random weights assigned
coid(Net,weighted=TRUE)

##input as binary matrices, with row names.
set.seed(12)
```

```

md1 <- matrix(sample(c(0,1),8*11,replace=TRUE),8,11,
               dimnames = list(paste0("b",1:8),paste0("c",1:11)))
md2 <- matrix(sample(c(0,1),10*12,replace=TRUE),10,12,
               dimnames = list(paste0("b",1:10),paste0("a",1:12)))
coid(md1,md2)

##input as weighted matrices,with row numbers as row names.
set.seed(12)
mdw1 <- matrix(sample(c(rep(0,40),runif(48,0,1))),8,11)
mdw2 <- matrix(sample(c(rep(0,40),runif(80,0,1))),10,12)
coid(mdw1,mdw2)
coid(mdw1,mdw2,weighted=TRUE)
coid(mdw1,mdw2,weighted=TRUE, weight_type="sum")

```

cois

*Correlation of Interaction Similarity for connector nodes***Description**

Calculating correlation of interaction similarity for connector nodes.

**Usage**

```

cois(
  network.or.subnet_mat1,
  subnet_mat2 = NULL,
  weighted = FALSE,
  method = "kendall"
)

```

**Arguments**

network.or.subnet_mat1	An igraph object or matrix. An "igraph" object with node attribute 'level' or a matrix representing one subnetwork. See details.
subnet_mat2	A matrix representing one subnetwork.
weighted	Logical. Default to FALSE. See details.
method	Correlation method ("pearson", "kendall" or "spearman"). Default to "kendall".

**Details**

In this package, a tripartite network contains three groups of nodes (a-nodes,b-nodes,c-nodes) and two subnetworks (P includes the links between a-nodes and b-nodes, Q includes the links between b-nodes and c-nodes). Connector nodes belong to b-nodes.

This function follows Sauve et al.(2016) to calculate the correlation of interaction similarity of connector nodes. When weighted=FALSE, the Jaccard similarity coefficients between all pairs of connector nodes in each subnetwork, defined as the number of shared interaction partners divided

by the total number of interaction partners involved (i.e. the maximum number of interactions they could share). When `weighted=TRUE`, the quantitative similarities of all pairs of connector nodes in each subnetwork were measured using the generalised Jaccard coefficient. Three correlation methods are provided. Kendall correlation is recommended following Sauve et al.(2016).

Two types of inputs `network` or `subnet_mat1` can be processed:

- An "igraph" object with node attribute 'level' (0 for a-nodes, 1 for b-nodes,2 for c-nodes). If the input is a weighted network, the edge should have a 'weight' attribute.
- Or a matrix representing subnetwork P, and must be input with `subnet_mat2` representing subnetwork Q.

If the inputs are two matrices, please make sure the rows of `network` or `subnet_mat1` and `subnet_mat2` both represent the groups of connector species,i.e, the b-group species. If both matrices have row names, then the function matches row names to produce connector nodes. Otherwise, row numbers are assigned to row names and matched. Within the two matrices (P and Q), columns represents a-group nodes and c-group nodes respectively. Elements in matrices are non-zero values if two nodes are linked with or without weights, and 0 otherwise.

### Value

Return a numeric value representing correlation of interaction similarity for connector species.

### References

Sauve, A. M., Thebault, E., Poccock, M. J., & Fontaine, C. (2016). How plants connect pollination and herbivory networks and their contribution to community stability. *Ecology*, 97(4), 908-917.

### Examples

```
## generate a random binary tripartite network
set.seed(12)
Net <- build_toy_net(11,15,16,0.2)
cois(Net)
## empirical network
data(PPH_Coltparkmeadow)
Net <- PPH_Coltparkmeadow
cois(Net)
set.seed(13)
library(igraph)
E(Net)$weight<-runif(length(E(Net)),0.1,1.5)#random weights assigned
cois(Net,weighted=TRUE)

##input as binary matrices,with row names.
md1 <- matrix(sample(c(0,1),8*11,replace=TRUE),8,11,
  dimnames = list(paste0("b",1:8),paste0("c",1:11)))
md2 <- matrix(sample(c(0,1),10*12,replace=TRUE),10,12,
  dimnames = list(paste0("b",1:10),paste0("a",1:12)))
cois(md1,md2)

##input as weighted matrices,with row numbers as row names.
mdw1 <- matrix(sample(c(rep(0,40),runif(48,0,1))),8,11)
```

```
mdw2 <- matrix(sample(c(rep(0,40),runif(80,0,1))),10,12)
cois(mdw1,mdw2)
cois(mdw1,mdw2,weighted=TRUE)
```

---

hc *The proportion of hub connectors in the shared sets of nodes*

---

## Description

This function calculates the proportion of shared node hubs that are connectors (HC), i.e. the top  $x\%$  of the nodes in the shared set of nodes with the highest degree that are connector nodes.

## Usage

```
hc(network.or.subnet_mat1, subnet_mat2 = NULL, hubs = 0.2, weighted = FALSE)
```

## Arguments

network.or.subnet_mat1	An igraph object or matrix. An "igraph" object with node attribute 'level' or a matrix representing one subnetwork. See details.
subnet_mat2	A matrix representing one subnetwork.
hubs	The top $x\%$ of the nodes in the shared set of nodes with the highest degree. default to 20%.
weighted	Logical. Default to FALSE. If TRUE, weighted degree defined as the sum of link strengths attached to a connector node is used.

## Details

In this package, a tripartite network contains three groups of nodes (a-nodes,b-nodes,c-nodes) and two subnetworks (P includes the links between a-nodes and b-nodes, Q includes the links between b-nodes and c-nodes). Connector nodes belong to b-nodes.

The function counts the proportion of connector nodes in connector node hubs (HC). The connector node hubs are the top  $x\%$  of shared nodes with the highest degree (Dominguez-Garcia and Kefi 2024). The default  $x\%$  is 20%. It always equals 1 if all b-nodes are connector nodes.

Two types of inputs network.or.subnet\_mat1 can be processed:

- An "igraph" object with node attribute 'level' (0 for a-nodes, 1 for b-nodes, 2 for c-nodes). If the input is a weighted network, the edge should have a 'weight' attribute.
- Or a matrix representing subnetwork P, and must be input with subnet\_mat2 representing subnetwork Q.

If the inputs are two matrices, please make sure the rows of network.or.subnet\_mat1 and subnet\_mat2 both represent the groups of connector species,i.e, the b-group species. If both matrices have row names, then the function matches row names to produce connector nodes. Otherwise, row numbers are assigned to row names and matched. Within the two matrices (P and Q), columns represents a-group nodes and c-group nodes respectively. Elements in matrices are non-zero values if two nodes are linked with or without weights, and 0 otherwise.

**Value**

Return a numeric value representing the proportion of connector nodes in node hubs.

**References**

Battiston, F., Nicosia, V. & Latora, V. (2014) Structural measures for multiplex networks. *Physical Review E*, 89, 032804.

Dominguez-Garcia, V. and Kefi, S. (2024). The structure and robustness of ecological networks with two interaction types. *PLOS Computational Biology*, 20(1), e1011770.

Guimera, R. & Amaral, L.A.N. (2005) Cartography of complex networks: modules and universal roles. *Journal of Statistical Mechanics: Theory and Experiment*, 2005, P02001.

**Examples**

```
## generate a random binary tripartite network
set.seed(12)
Net <- build_toy_net(11,15,16,0.2)
hc(Net)

#empirical network
data(PPH_Coltparkmeadow)
Net <- PPH_Coltparkmeadow
hc(Net)
set.seed(13)
library(igraph)
E(Net)$weight<-runif(length(E(Net)),0.1,1)#random weights assigned
pc(Net,weighted=TRUE)

##input as binary matrices,with row names.
set.seed(12)
md1 <- matrix(sample(c(0,1),5*8,replace=TRUE),5,8)
dimnames(md1) = list(paste0("b",1:5),paste0("c",1:8))
md2 <- matrix(sample(c(0,1),20*30,replace=TRUE),20,30)
dimnames(md2) = list(paste0("b",1:20),paste0("a",1:30))
hc(md1,md2)

##input as weighted matrices,with row numbers as row names.
set.seed(17)
mdw1 <- matrix(sample(c(rep(0,20),runif(20,0,1))),5,8)
mdw2 <- matrix(sample(c(rep(0,500),runif(100,0,1))),20,30)
hc(mdw1,mdw2)
hc(mdw1,mdw2,weighted=TRUE)
```



**Description**

Counting the frequencies of interconnection motifs for a tripartite interaction network.

**Usage**

```
icmotif_count(network.or.subnet_mat1, subnet_mat2 = NULL, weighted = FALSE)
```

**Arguments**

network.or.subnet_mat1	An igraph object or matrix. An "igraph" object with node attribute 'level' or a matrix representing one subnetwork. See details.
subnet_mat2	A matrix representing one subnetwork.
weighted	Logical. Default to FALSE. If TRUE, the arithmetic mean of the subgraph weights is provided for each motif. See details

**Details**

In this package, a tripartite network contains three groups of nodes (a-nodes,b-nodes,c-nodes) and two subnetworks (P includes the links between a-nodes and b-nodes, Q includes the links between b-nodes and c-nodes). Connector nodes belong to b-nodes. An interconnection motif is defined to comprise three sets of connected nodes: the connector nodes (belonging to b-nodes), the nodes in one subnetwork (belonging to a-nodes in the P subnetwork), and the nodes in the other subnetwork (belonging to c-nodes in the Q subnetwork). Each motif has maximumly 6 nodes, resulting in a total of 48 distinct motif forms. The algorithm for counting interconnection motifs is designed by extending the fast approach from Simmons et al.(2019), which uses mathematical operations directly on the bi-adjacency matrix. For interconnection motifs in tripartite networks with intra-guild interactions, please see **ig\_icmotif\_count** and **ig\_icmotif\_role**.

Two types of inputs network.or.subnet\_mat1 can be processed:

- An "igraph" object with node attribute 'level' (0 for a-nodes, 1 for b-nodes,2 for c-nodes). If the input is a weighted network, the edge should have a 'weight' attribute.
- Or a matrix representing subnetwork P, and must be input with subnet\_mat2 representing subnetwork Q.

If the inputs are two matrices, please make sure the rows of network.or.subnet\_mat1 and subnet\_mat2 both represent the groups of connector species,i.e, the b-group species. If both matrices have row names, then the function matches row names to produce connector nodes. Otherwise, row numbers are assigned to row names and matched. Within the two matrices (P and Q), columns represents a-group nodes and c-group nodes respectively. Elements in matrices are non-zero values if two nodes are linked with or without weights, and 0 otherwise.

**Weighted networks** For weighted tripartite networks, the mean weight of a given motif is provided by averaging the weights of all the cases of a particular motif. The weight of a motif case is the arithmetic mean of the weights of its links, following Mora et al. (2018) and Simmons et al. (2019).

**Value**

Return a data.frame of the frequencies (and mean weight) of 48 interconnection motifs. See 'Multi\_motif' for the forms.

## References

- Pilosof, S., Porter, M. A., Pascual, M., & KC)fi, S. (2017). The multilayer nature of ecological networks. *Nature Ecology & Evolution*, 1(4), 0101.
- Mora, B.B., Cirtwill, A.R. and Stouffer, D.B. (2018). pymfinder: a tool for the motif analysis of binary and quantitative complex networks. *bioRxiv*, 364703.
- Simmons, B. I., Sweering, M. J., Schillinger, M., Dicks, L. V., Sutherland, W. J., & Di Clemente, R. (2019). bmotif: A package for motif analyses of bipartite networks. *Methods in Ecology and Evolution*, 10(5), 695-701.

## Examples

```
## generate a random tripartite network
set.seed(12)
Net <- build_toy_net(11,15,16,0.2)
icmotif_count(Net)

## empirical network
data(PPH_Coltparkmeadow)
Net <- PPH_Coltparkmeadow
icmotif_count(Net)
set.seed(13)
library(igraph)
E(Net)$weight<-runif(length(E(Net)),0.1,1)#random weights assigned
icmotif_count(Net, weighted=TRUE)

##input as binary matrices, with row names.
set.seed(12)
md1 <- matrix(sample(c(0,1),8*11,replace=TRUE),8,11)
dimnames(md1) = list(paste0("b",1:8),paste0("c",1:11))
md2 <- matrix(sample(c(0,1),10*12,replace=TRUE),10,12)
dimnames(md2) = list(paste0("b",1:10),paste0("a",1:12))
icmotif_count(md1,md2)

##input as weighted matrices,with row numbers as row names.
set.seed(12)
mdw1 <- matrix(sample(c(rep(0,40),runif(48,0,1))),8,11)
mdw2 <- matrix(sample(c(rep(0,40),runif(80,0,1))),10,12)
icmotif_count(mdw1,mdw2,weighted=TRUE)
```

---

icmotif\_role

*Count the roles of connector nodes defined by interconnection motifs*

---

## Description

Count the roles of connector nodes defined by interconnection motifs in a tripartite network.

**Usage**

```
icmotif_role(network.or.subnet_mat1, subnet_mat2 = NULL, weighted = FALSE)
```

**Arguments**

network.or.subnet_mat1	An igraph object or matrix. An "igraph" object with node attribute 'level' or a matrix representing one subnetwork. See details.
subnet_mat2	A matrix representing one subnetwork.
weighted	Logical. Default to FALSE. If TRUE, a weighted measure is provided. See details.

**Details**

In this package, a tripartite network contains three groups of nodes (a-nodes,b-nodes,c-nodes) and two subnetworks (P includes the links between a-nodes and b-nodes, Q includes the links between b-nodes and c-nodes). Connector nodes belong to b-nodes. An interconnection motif is defined to comprise three sets of connected nodes: the connector nodes (belonging to b-nodes), the nodes in one subnetwork (belonging to a-nodes in the P subnetwork), and the nodes in the other subnetwork (belonging to c-nodes in the Q subnetwork). Each motif has maximumly 6 nodes, resulting in a total of 48 distinct motif forms. The algorithm for counting interconnection motifs is designed by extending the fast approach from Simmons et al.(2019), which uses mathematical operations directly on the bi-adjacency matrix. For interconnection motifs in tripartite networks with intra-guild interactions, please see **ig\_icmotif\_count** and **ig\_icmotif\_role**.

Two types of inputs network.or.subnet\_mat1 can be processed:

- An "igraph" object with node attribute 'level' (0 for a-nodes, 1 for b-nodes,2 for c-nodes). If the input is a weighted network, the edge should have a 'weight' attribute.
- Or a matrix representing subnetwork P, and must be input with subnet\_mat2 representing subnetwork Q.

If the inputs are two matrices, please make sure the rows of network.or.subnet\_mat1 and subnet\_mat2 both represent the groups of connector species,i.e, the b-group species. If both matrices have row names, then the function matches row names to produce connector nodes. Otherwise, row numbers are assigned to row names and matched. Within the two matrices (P and Q), columns represents a-group nodes and c-group nodes respectively. Elements in matrices are non-zero values if two nodes are linked with or without weights, and 0 otherwise.

**Weighted networks** For weighted tripartite networks, the mean weight of the motif occurrence (i.e., a motif occurrence isomorphic to a particular motif form) is provided for a given node with a given role, following Mora et al. (2018) and Simmons et al. (2019).

**Value**

For binary networks, return a matrix with elements representing the number of times each connector node plays for each unique role within interconnection motifs; for weighted networks, the matrix element represents the mean weight of the motif occurrences where the node exists.

## References

# Mora, B.B., Cirtwill, A.R. and Stouffer, D.B. (2018). pymfinder: a tool for the motif analysis of binary and quantitative complex networks. bioRxiv, 364703.

Simmons, B. I., Sweering, M. J., Schillinger, M., Dicks, L. V., Sutherland, W. J., & Di Clemente, R. (2019). bmotif: A package for motif analyses of bipartite networks. *Methods in Ecology and Evolution*, 10(5), 695-701.

## Examples

```
## generate a random tripartite network
set.seed(12)
Net <- build_toy_net(11,15,16,0.2)
icmotif_role(Net)

## empirical network
data(PPH_Coltparkmeadow)
Net <- PPH_Coltparkmeadow
icmotif_role(Net)
set.seed(13)
library(igraph)
E(Net)$weight<-runif(length(E(Net)),0.1,1)#random weights assigned
icmotif_role(Net, weighted=TRUE)

##input as binary matrices, with row names.
set.seed(12)
md1 <- matrix(sample(c(0,1),8*11,replace=TRUE),8,11)
dimnames(md1) = list(paste0("b",1:8),paste0("c",1:11))
md2 <- matrix(sample(c(0,1),10*12,replace=TRUE),10,12)
dimnames(md2) = list(paste0("b",1:10),paste0("a",1:12))
icmotif_role(md1,md2)

##input as weighted matrices,with row numbers as row names.
set.seed(12)
mdw1 <- matrix(sample(c(rep(0,40),runif(48,0,1))),8,11)
mdw2 <- matrix(sample(c(rep(0,40),runif(80,0,1))),10,12)
icmotif_role(mdw1,mdw2,weighted=TRUE)
```

---

ig\_ddom

*Degree of diagonal dominance*

---

## Description

Calculating the degree of diagonal dominance for a tripartite network with intra-guild interactions.

## Usage

```
ig_ddom(mat)
```

**Arguments**

`mat` A square block interaction matrix representing a tripartite network including intra-guild and inter-guild interactions. See details.

**Details**

To calculate degree of diagonal dominance, the input should be a block matrix ( $M$ ) to represent three groups of nodes (a-nodes, b-nodes and c-nodes): three intra-guild interaction matrices ( $m_{aa}, m_{bb}, m_{cc}$ ), two inter-guild matrices of a and b-nodes ( $m_{ab}, m_{ba}$ ), and two inter-guild matrices of b- and c-nodes ( $m_{bc}, m_{cb}$ ).

$$\begin{pmatrix} m_{aa} & m_{ab} & 0 \\ m_{ba} & m_{bb} & m_{bc} \\ 0 & m_{cb} & m_{cc} \end{pmatrix}$$

This function follows the definition by Garcia-Callejas et al (2023). Diagonal dominance is a matrix property indicating diagonal elements are larger than the sum of non-diagonal elements. In a tripartite network with intra-guild interactions, Garcia-Callejas et al. propose a continuous diagonal dominance index by calculating the difference between the diagonal and the sum of non-diagonal elements for each node. The average degree of diagonal dominance ( $d$ ) is calculated by averaging all the nodes' values.

$$d = \frac{1}{S} \sum_{i \in S} \left( M_{ii} - \sum_{\substack{j \in S \\ j \neq i}} M_{ij} \right)$$

Where  $S$  is the number of species in the tripartite network  $M$ .

**Value**

Return the average degree of diagonal dominance and a vector of diagonal dominance indices for all nodes.

**References**

Garcia-Callejas, D., Godoy, O., Buche, L., Hurtado, M., Lanuza, J.B., Allen-Perkins, A. et al. (2023) Non-random interactions within and across guilds shape the potential to coexist in multi-trophic ecological communities. *Ecology Letters*, 26, 831-842.

**Examples**

```
##A toy tripartite network with intra-guild negative interactions:
##Inter-guild mutualistic interactions and inter-guild antagonistic interactions.
set.seed(12)
## 4 a-nodes, 5 b-nodes, and 3 c-nodes.

##Intra-guild interaction matrices
mat_aa<-matrix(runif(16,-0.8,-0.2),4,4)
mat_bb<-matrix(runif(25,-0.8,-0.2),5,5)
mat_cc<-matrix(runif(9,-0.8,-0.2),3,3)
```

```

##Inter-guild interaction matrices between a- and b-nodes.
mat_ab<-mat_ba<-matrix(sample(c(rep(0,8),runif(12,0,0.5))),4,5,byrow=TRUE)
mat_ba[mat_ba>0]<-runif(12,0,0.5);mat_ba<-t(mat_ba)

##inter-guild interaction matrices between b- and c-nodes.
mat_cb<-mat_bc<-matrix(sample(c(rep(0,8),runif(7,0,0.5))),3,5,byrow=TRUE)
mat_bc[mat_bc>0]<-runif(7,0,0.5);mat_bc<-t(mat_bc)
mat<-rbind(cbind(mat_aa,mat_ab,matrix(0,4,3)),cbind(mat_ba,mat_bb,mat_bc))
mat<-rbind(mat,cbind(matrix(0,3,4),mat_cb,mat_cc))

##set the node names
rownames(mat)<-c(paste0("a",1:4),paste0("b",1:5),paste0("c",1:3))
colnames(mat)<-c(paste0("a",1:4),paste0("b",1:5),paste0("c",1:3))
diag(mat)<--1 #assume -1 for diagonal elements

##Visualization of the block matrix.
library(plot.matrix)
pal <- colorRampPalette(c("darkblue", "lightblue", "white", "pink", "darkred"))(100)
par(mar=c(5,5,5,5));plot(mat,col = pal,
    breaks = seq(-max(abs(mat)), max(abs(mat)), length.out = 101),
    main = "Matrix visualization")
clip(x1 = 0.5,# Left boundary
     x2 = ncol(mat) + 0.5, # Right boundary
     y1 = 0.5,           # Top boundary
     y2 = nrow(mat) + 0.5 )
abline(v = c(4.5,9.5), h = c(3.5,8.5), lwd = 3, col = "black")

ig_ddom(mat)

```

---

ig_icmotif_count	<i>Count interconnection motifs for tripartite networks with intra-guild interactions</i>
------------------	---

---

## Description

Counting the frequencies of interconnection motifs for a tripartite interaction network with intra-guild interactions.

## Usage

```
ig_icmotif_count(mat, guilds, weighted = FALSE)
```

## Arguments

mat	A square block interaction matrix representing a tripartite network including intra-guild and inter-guild interactions. See details.
guilds	A character vector matching rows of mat to indicate the guilds using ('a','b' and 'c'). See details.
weighted	Logical. Default to FALSE. If TRUE, the arithmetic mean of the subgraph weights is provided for each motif. See details

## Details

This function is designed for tripartite networks with intra-guild interactions. The input network should be nput as a block matrix ( $M$ ) to represent three groups of nodes (a-nodes, b-nodes and c-nodes): three intra-guild interaction matrices ( $m_{aa}, m_{bb}, m_{cc}$ ), two inter-guild matrices of a and b-nodes ( $m_{ab}, m_{ba}$ ), and two inter-guild matrices of b- and c-nodes ( $m_{bc}, m_{cb}$ ).

$$\begin{pmatrix} m_{aa} & m_{ab} & 0 \\ m_{ba} & m_{bb} & m_{bc} \\ 0 & m_{cb} & m_{cc} \end{pmatrix}$$

guilds should be a vector of the same length as the row of mat like c("a","a"... "b","b"... "c","c"..)

**Interconnection motifs in tripartite networks with intra-guild interactions** An interconnection motif is defined to comprise three sets of connected nodes: the connector nodes (belonging to b-nodes), the nodes in one subnetwork (belonging to a-nodes in the P subnetwork), and the nodes in the other subnetwork (belonging to c-nodes in the Q subnetwork). Each guild has two nodes at most, resulting in a total of 107 distinct motif forms. The algorithm for counting interconnection motifs is designed by extending the fast approach from Simmons et al.(2019). For interconnection motifs in tripartite networks without intra-guild interactions, please see **icmotif\_count** and **icmotif\_role**.

**Weighted networks** For weighted tripartite networks, the mean weight of a given motif is provided by averaging the weights of all motif occurrences isomorphic to the motif. The weight of a motif occurrence is the arithmetic mean of the weights of its links, following Mora et al. (2018) and Simmons et al. (2019).

## Value

Return a matrix of counts (and mean weight) of 107 interconnection motifs.

## References

- Garcia-Callejas, D., Godoy, O., Buche, L., Hurtado, M., Lanuza, J.B., Allen-Perkins, A. et al. (2023) Non-random interactions within and across guilds shape the potential to coexist in multi-trophic ecological communities. *Ecology Letters*, 26, 831-842.
- Pilosof, S., Porter, M. A., Pascual, M., & KC)fi, S. (2017). The multilayer nature of ecological networks. *Nature Ecology & Evolution*, 1(4), 0101.
- Mora, B.B., Cirtwill, A.R. and Stouffer, D.B. (2018). pymfinder: a tool for the motif analysis of binary and quantitative complex networks. *bioRxiv*, 364703.
- Simmons, B. I., Sweering, M. J., Schillinger, M., Dicks, L. V., Sutherland, W. J., & Di Clemente, R. (2019). bmotif: A package for motif analyses of bipartite networks. *Methods in Ecology and Evolution*, 10(5), 695-701.

## Examples

```
##A toy tripartite network with intra-guild negative interactions:
##Inter-guild mutualistic interactions and inter-guild antagonistic interactions.
set.seed(12)
##4 a-nodes, 5 b-nodes, and 3 c-nodes

##Intra-guild interaction matrices
```

```

mat_aa<-matrix(runif(16,-0.8,-0.2),4,4)
mat_aa <- mat_aa + t(mat_aa); diag(mat_aa) <- 0
mat_bb<-matrix(runif(25,-0.8,-0.2),5,5)
mat_bb <- mat_bb + t(mat_bb); diag(mat_bb) <- 0
mat_cc<-matrix(runif(9,-0.8,-0.2),3,3)
mat_cc <- mat_cc + t(mat_cc); diag(mat_cc) <- 0

##Inter-guild interaction matrices between a- and b-nodes.
mat_ab<-mat_ba<-matrix(sample(c(rep(0,8),runif(12,0,0.5))),4,5,byrow=TRUE)
mat_ba<-t(mat_ba)

##Inter-guild interaction matrices between b- and c-nodes.
mat_cb<-mat_bc<-matrix(sample(c(rep(0,8),runif(7,0,0.5))),3,5,byrow=TRUE)
mat_bc<--t(mat_bc)
mat<-rbind(cbind(mat_aa,mat_ab,matrix(0,4,3)),cbind(mat_ba,mat_bb,mat_bc))
mat<-rbind(mat,cbind(matrix(0,3,4),mat_cb,mat_cc))

##set the node names
rownames(mat)<-c(paste0("a",1:4),paste0("b",1:5),paste0("c",1:3))
colnames(mat)<-c(paste0("a",1:4),paste0("b",1:5),paste0("c",1:3))

#mat[mat!=0] <- 1
myguilds=c(rep("a",4),rep("b",5),rep("c",3))
ig_icmotif_count(mat,guilds=myguilds,TRUE)

```

---

ig_icmotif_role	<i>Count the roles of connector nodes defined by interconnection motifs with intra-guild interactions</i>
-----------------	---

---

## Description

Count the roles of connector nodes defined by interconnection motifs in a tripartite network with intra-guild interactions.

## Usage

```
ig_icmotif_role(mat, guilds, weighted = FALSE)
```

## Arguments

mat	A square block interaction matrix representing a tripartite network including intra-guild and inter-guild interactions. See details.
guilds	A character vector matching rows of mat to indicate the guilds using ('a','b' and 'c'). See details.
weighted	Logical. Default to FALSE. If TRUE, a weighted measure is provided. See details.



## Details

This function is designed for tripartite networks with intra-guild interactions. The input network should be nput as a block matrix ( $M$ ) to represent three groups of nodes (a-nodes, b-nodes and c-nodes): three intra-guild interaction matrices ( $m_{aa}, m_{bb}, m_{cc}$ ), two inter-guild matrices of a and b-nodes ( $m_{ab}, m_{ba}$ ), and two inter-guild matrices of b- and c-nodes ( $m_{bc}, m_{cb}$ ).

$$\begin{pmatrix} m_{aa} & m_{ab} & 0 \\ m_{ba} & m_{bb} & m_{bc} \\ 0 & m_{cb} & m_{cc} \end{pmatrix}$$

guilds should be a vector of the same length as the row of mat like c("a","a"... "b","b"... "c","c"..)

**Interconnection motifs in tripartite networks with intra-guild interactions** An interconnection motif is defined to comprise three sets of connected nodes: the connector nodes (belonging to b-nodes), the nodes in one subnetwork (belonging to a-nodes in the P subnetwork), and the nodes in the other subnetwork (belonging to c-nodes in the Q subnetwork). Each guild has two nodes at most, resulting in a total of 107 distinct motif forms. The algorithm for counting interconnection motifs is designed by extending the fast approach from Simmons et al.(2019). Currently, only the roles in first 35 motifs are provided (see ). For interconnection motifs in tripartite networks without intra-guild interactions, please see **icmotif\_count** and **icmotif\_role**.

**Weighted networks** For weighted tripartite networks, the mean weight of the motif occurrence (i.e., a motif occurrence isomorphic to a particular motif form) is provided for a given node with a given role, following Mora et al. (2018) and Simmons et al. (2019).

## Value

For binary networks, return a matrix with elements representing the number of times each connector node plays for each unique role within interconnection motifs; for weighted networks, the matrix element represents the mean weight of the motif occurrences where the node exists.

## References

- Garcia-Callejas, D., Godoy, O., Buche, L., Hurtado, M., Lanuza, J.B., Allen-Perkins, A. et al. (2023) Non-random interactions within and across guilds shape the potential to coexist in multi-trophic ecological communities. *Ecology Letters*, 26, 831-842.
- Pilosof, S., Porter, M. A., Pascual, M., & KC)fi, S. (2017). The multilayer nature of ecological networks. *Nature Ecology & Evolution*, 1(4), 0101.
- Mora, B.B., Cirtwill, A.R. and Stouffer, D.B. (2018). pymfinder: a tool for the motif analysis of binary and quantitative complex networks. *bioRxiv*, 364703.
- Simmons, B. I., Sweering, M. J., Schillinger, M., Dicks, L. V., Sutherland, W. J., & Di Clemente, R. (2019). bmotif: A package for motif analyses of bipartite networks. *Methods in Ecology and Evolution*, 10(5), 695-701.

## Examples

```
##A toy tripartite network with intra-guild negative interactions:
##Inter-guild mutualistic interactions and inter-guild antagonistic interactions.
set.seed(12)
##4 a-nodes, 5 b-nodes, and 3 c-nodes
```

```

##Intra-guild interaction matrices
mat_aa<-matrix(runif(16,-0.8,-0.2),4,4)
mat_aa <- mat_aa + t(mat_aa); diag(mat_aa) <- 0
mat_bb<-matrix(runif(25,-0.8,-0.2),5,5)
mat_bb <- mat_bb + t(mat_bb); diag(mat_bb) <- 0
mat_cc<-matrix(runif(9,-0.8,-0.2),3,3)
mat_cc <- mat_cc + t(mat_cc); diag(mat_cc) <- 0

##Inter-guild interaction matrices between a- and b-nodes.
mat_ab<-mat_ba<-matrix(sample(c(rep(0,8),runif(12,0,0.5))),4,5,byrow=TRUE)
mat_ba<-t(mat_ba)

##Inter-guild interaction matrices between b- and c-nodes.
mat_cb<-mat_bc<-matrix(sample(c(rep(0,8),runif(7,0,0.5))),3,5,byrow=TRUE)
mat_bc<--t(mat_bc)
mat<-rbind(cbind(mat_aa,mat_ab,matrix(0,4,3)),cbind(mat_ba,mat_bb,mat_bc))
mat<-rbind(mat,cbind(matrix(0,3,4),mat_cb,mat_cc))

##Set the node names
rownames(mat)<-c(paste0("a",1:4),paste0("b",1:5),paste0("c",1:3))
colnames(mat)<-c(paste0("a",1:4),paste0("b",1:5),paste0("c",1:3))

myguilds=c(rep("a",4),rep("b",5),rep("c",3))
ig_icmotif_role(mat,guilds=myguilds)
ig_icmotif_role(mat,guilds=myguilds,TRUE)

```

---

ig\_overlap\_guild      *Intra-guild and inter\_guild interaction overlap*

---

### Description

Calculating species-level intra-guild and inter\_guild interaction overlap for a tripartite network with intra-guild interactions.

### Usage

```
ig_overlap_guild(mat, guilds, method = "horn")
```

### Arguments

mat	A square block interaction matrix representing a tripartite network including intra-guild and inter-guild interactions. See details.
guilds	A character vector matching rows of mat to indicate the guilds using ('a','b' and 'c'). See details.
method	The distance method. Same with <i>vegan::vegdist</i> . Default to "horn"

## Details

The input is a block matrix ( $M$ ) to represent interactions among three groups of species (a-nodes, b-nodes and c-nodes): three intra-guild interaction matrices ( $m_{aa}, m_{bb}, m_{cc}$ ), two inter-guild matrices of a and b-nodes ( $m_{ab}, m_{ba}$  with symmetric links), and two inter-guild matrices of b- and c-nodes ( $m_{bc}, m_{cb}$  with symmetric links). Connector species belong to b-nodes.

$$\begin{pmatrix} m_{aa} & m_{ab} & 0 \\ m_{ba} & m_{bb} & m_{bc} \\ 0 & m_{cb} & m_{cc} \end{pmatrix}$$

guilds should be a vector of the same length as the row of mat like c("a","a"... "b","b"... "c","c"..)

This function follows the definition by Garcia-Callejas et al (2023). Species-level interaction overlap is derived from the overlap between each pair of species, calculated using the dissimilarity index ( $d_{ij}$ , default to Morisita-Horn index) as in the R package *vegan*. The net overlap of species ( $o_i$ ) is represented by the sum of pairwise overlaps with every other species:

$$o_i = \sum_{j \in S} (1 - d_{ij})$$

## Value

Return a list including three species-level intra-guild overlap vectors for a-, b- and c-nodes (a\_intra\_overlap, b\_intra\_overlap, c\_intra\_overlap), two vectors of inter-guild interaction overlap for a-nodes (a\_inter\_b\_overlap) and b-nodes (b\_inter\_a\_overlap), and two vectors of inter-guild interaction overlap for b-nodes (b\_inter\_c\_overlap) and c-nodes (c\_inter\_b\_overlap).

## References

Garcia-Callejas, D., Godoy, O., Buche, L., Hurtado, M., Lanuza, J.B., Allen-Perkins, A. et al. (2023) Non-random interactions within and across guilds shape the potential to coexist in multi-trophic ecological communities. *Ecology Letters*, 26, 831-842.

## Examples

```
##A toy tripartite network with intra-guild negative interactions,
##Inter-guild mutualistic interactions and inter-guild antagonistic interactions.
set.seed(12)
##4 a-nodes, 5 b-nodes, and 3 c-nodes

##Intra-guild interaction matrices
mat_aa<-matrix(runif(16,-0.8,-0.2),4,4)
mat_bb<-matrix(runif(25,-0.8,-0.2),5,5)
mat_cc<-matrix(runif(9,-0.8,-0.2),3,3)

##Inter-guild interaction matrices between a- and b-nodes.
mat_ab<-mat_ba<-matrix(sample(c(rep(0,8),runif(12,0,0.5))),4,5,byrow=TRUE)
mat_ba[mat_ba>0]<-runif(12,0,0.5);mat_ba<-t(mat_ba)

##Inter-guild interaction matrices between b- and c-nodes.
mat_cb<-mat_bc<-matrix(sample(c(rep(0,8),runif(7,0,0.5))),3,5,byrow=TRUE)
mat_bc[mat_bc>0]<-runif(7,0,0.5);mat_bc<-t(mat_bc)
```

```

mat<-rbind(cbind(mat_aa,mat_ab,matrix(0,4,3)),cbind(mat_ba,mat_bb,mat_bc))
mat<-rbind(mat,cbind(matrix(0,3,4),mat_cb,mat_cc))

##Set the node names
rownames(mat)<-c(paste0("a",1:4),paste0("b",1:5),paste0("c",1:3))
colnames(mat)<-c(paste0("a",1:4),paste0("b",1:5),paste0("c",1:3))
diag(mat)<--1 #assume -1 for diagonal elements

##Visualization of this block matrix.
library(plot.matrix)
pal <- colorRampPalette(c("darkblue", "lightblue", "white", "pink", "darkred"))(100)
par(mar=c(5,5,5,5));plot(mat,col = pal,
      breaks = seq(-max(abs(mat)), max(abs(mat)), length.out = 101),
      main = "Matrix visualization")
clip(x1 = 0.5,# Left boundary
     x2 = ncol(mat) + 0.5, # Right boundary
     y1 = 0.5,           # Top boundary
     y2 = nrow(mat) + 0.5 )
abline(v = c(4.5,9.5), h = c(3.5,8.5), lwd = 3, col = "black")

myguilds=c(rep("a",4),rep("b",5),rep("c",3))
ig_overlap_guild(mat,guilds=myguilds)

```

---

Multi\_motif

*Interconnection motif forms*


---

## Description

The dictionary of interconnection motif forms without intra-guild interactions.

## Usage

```

Multi_motif(
  ID = c("M111", "M112", "M113", "M114", "M211", "M212", "M213", "M311", "M312", "M411",
        "M121", "M122-1", "M122-2", "M122-3", "M123-1", "M123-2", "M123-3", "M123-4",
        "M123-5", "M221-1", "M221-2", "M221-3", "M222-1", "M222-2", "M222-3", "M222-4",
        "M222-5", "M222-6", "M222-7", "M222-8", "M222-9", "M321-1", "M321-2", "M321-3",
        "M321-4", "M321-5", "M131", "M132-1", "M132-2", "M132-3", "M132-4", "M132-5",
        "M231-1", "M231-2", "M231-3", "M231-4", "M231-5", "M141", "all")
)

```

## Arguments

ID                      Character. Default to "all". See details for the IDs representing different motifs.

## Details

An interconnection motif is defined to comprise three sets of connected nodes: the connector nodes (belonging to b-nodes), the nodes in one subnetwork (belonging to a-nodes in the P subnetwork), and the nodes in the other subnetwork (belonging to c-nodes in the Q subnetwork). Each motif has maximumly 6 nodes, resulting in a total of 48 distinct motif forms.

**Value**

If ID is one ID, it returns a corresponding motif "igraph" object.

If ID is "all", it returns a list containing all 48 motifs of "igraph" objects.

**Examples**

```
ID<-c("M111", "M112", "M113", "M114", "M211", "M212", "M213", "M311",
      "M312", "M411", "M121", "M122-1", "M122-2", "M122-3", "M123-1",
      "M123-2", "M123-3", "M123-4", "M123-5", "M221-1", "M221-2",
      "M221-3", "M222-1", "M222-2", "M222-3", "M222-4", "M222-5",
      "M222-6", "M222-7", "M222-8", "M222-9", "M321-1", "M321-2",
      "M321-3", "M321-4", "M321-5", "M131", "M132-1", "M132-2",
      "M132-3", "M132-4", "M132-5", "M231-1", "M231-2", "M231-3",
      "M231-4", "M231-5", "M141")
```

```
m<-Multi_motif("M222-4")
plot(m,main="M222-4")
```

```
mr <- par(mfrow=c(4,6),mar=c(1,1,3,1))
for(i in 1:24){
  plot(Multi_motif("all")[[i]],
       vertex.size=30, vertex.label=NA,
       vertex.color="blue",main=ID[i])
}
par(mr)
```

---

node\_icc

*Interconnection centrality for connector nodes in a tripartite network*


---

**Description**

Calculating interconnection centrality (degree, betweenness and closeness) specified for connector nodes in a tripartite network.

**Usage**

```
node_icc(network.or.subnet_mat1, subnet_mat2 = NULL, weighted = FALSE)
```

**Arguments**

network.or.subnet\_mat1

An igraph object or matrix. An "igraph" object with node attribute 'level' or a matrix representing one subnetwork. See details.

subnet\_mat2 A matrix representing one subnetwork.

weighted Logical. If TRUE, link strengths of connector nodes are used. Default to FALSE.

## Details

In this package, a tripartite network contains three groups of nodes (a-nodes,b-nodes,c-nodes) and two subnetworks (P includes the links between a-nodes and b-nodes, Q includes the links between b-nodes and c-nodes). Connector nodes belong to b-nodes. This function calculates interconnection degree, betweenness and closeness centrality for connector nodes.

- For binary networks, **Interconnection degree** of each connector species is defined as the product of its degree values from two subnetworks.
- **Interconnection betweenness** is defined by the number of shortest paths from a-nodes to c-nodes going through connector species,  $\sum_{\substack{i \in a, j \in c \\ i \neq j}} \frac{g_{ivj}}{g_{ij}}$ , where  $g_{ij}$  is the total number of shortest paths between node  $i$  from a-nodes and  $j$  from c-nodes while  $g_{ivj}$  is the number of those shortest paths which pass through connector species  $v$ .
- **Interconnection closeness** is defined by the inverse of the sum of distances from connector species to both a-nodes and c-nodes,  $1/(\sum_{v \neq i, i \in a \cup c} d_{vi})$ , where  $d_{vi}$  is the distance between connector species  $v$  and species  $i$  from a-nodes and c-nodes. For weighted networks, interaction strengths are used in the calculation of weighted degree, shorest path, and distance.

### network.or.subnet\_mat1 and subnet\_mat2

Two types of inputs network.or.subnet\_mat1 can be processed:

- An "igraph" object with node attribute 'level' (0 for a-nodes, 1 for b-nodes,2 for c-nodes). If the input is a weighted network, the edge should have a 'weight' attribute.
- Or a matrix representing subnetwork P, and must be input with subnet\_mat2 representing subnetwork Q.

If the inputs are two matrices, please make sure the rows of network.or.subnet\_mat1 and subnet\_mat2 both represent the groups of connector species,i.e, the b-group species. If both matrices have row names, then the function matches row names to produce connector nodes. Otherwise, row numbers are assigned as row names.

## Value

Return a data frame with interconnection degree, betweenness and closeness for connector nodes.

## Examples

```
## generate a random tripartite network
set.seed(12)
Net <- build_toy_net(11,15,16,0.2)

data(PPH_Coltparkmeadow)
Net <- PPH_Coltparkmeadow
node_icc(Net)
set.seed(13)
library(igraph)
E(Net)$weight<-runif(length(E(Net)),0.1,1)#random weights assigned
node_icc(Net,weighted=TRUE)

##input as binary matrices,with row names.
```

```

set.seed(12)
md1 <- matrix(sample(c(0,1),5*8,replace=TRUE),5,8,
              dimnames = list(paste0("b",1:5),paste0("c",1:8)))
md2 <- matrix(sample(c(0,1),20*30,replace=TRUE),20,30,
              dimnames = list(paste0("b",1:20),paste0("a",1:30)))
node_icc(md1,md2)

##input as weighted matrices,with row numbers as row names.
set.seed(17)
mdw1 <- matrix(sample(c(rep(0,20),runif(20,0,1))),5,8)
mdw2 <- matrix(sample(c(rep(0,500),runif(100,0,1))),20,30)
node_icc(mdw1,mdw2)
node_icc(mdw1,mdw2,weighted=TRUE)

```

pc

*Participation coefficient***Description**

Calculating participation coefficient for connector nodes.

**Usage**

```
pc(network.or.subnet_mat1, subnet_mat2 = NULL, weighted = FALSE)
```

**Arguments**

network.or.subnet_mat1	An igraph object or matrix. An "igraph" object with node attribute 'level' or a matrix representing one subnetwork. See details.
subnet_mat2	A matrix representing one subnetwork.
weighted	Logical. Default to FALSE. If TRUE, weighted degree defined as the sum of interaction strengths attached to a connector node is used.

**Details**

In this package, a tripartite network contains three groups of nodes (a-nodes,b-nodes,c-nodes) and two subnetworks (P includes the links between a-nodes and b-nodes, Q includes the links between b-nodes and c-nodes). Connector nodes belong to b-nodes.

The **pc** function calculates the participation coefficient following Dominguez-Garcia and Kefi (2024). For each connector node  $i$ ,  $pc_i$  is calculated as two times the ratio between the lowest degree in both interaction subnetworks divided by the total degree of the node ( $2 \frac{d_{lowest}}{d_{total}}$ ). Hence, the participation coefficient for all connector nodes ( $PC_c$ ) is represented by the average value of all  $pc_i$ .

Two types of inputs network.or.subnet\_mat1 can be processed:

- An "igraph" object with node attribute 'level' (0 for a-nodes, 1 for b-nodes,2 for c-nodes). If the input is a weighted network, the edge should have a 'weight' attribute.

- Or a matrix representing subnetwork P, and must be input with `subnet_mat2` representing subnetwork Q.

If the inputs are two matrices, please make sure the rows of `network.or.subnet_mat1` and `subnet_mat2` both represent the groups of connector species, i.e, the b-group species. If both matrices have row names, then the function matches row names to produce connector nodes. Otherwise, row numbers are assigned to row names and matched. Within the two matrices (P and Q), columns represents a-group nodes and c-group nodes respectively. Elements in matrices are non-zero values if two nodes are linked with or without weights, and 0 otherwise.

### Value

Return a numeric value representing participation coefficient for connector species.

### References

Battiston, F., Nicosia, V. & Latora, V. (2014) Structural measures for multiplex networks. *Physical Review E*, 89, 032804.

Dominguez-Garcia, V. and Kefi, S. (2024). The structure and robustness of ecological networks with two interaction types. *PLOS Computational Biology*, 20(1), e1011770.

Guimera, R. & Amaral, L.A.N. (2005) Cartography of complex networks: modules and universal roles. *Journal of Statistical Mechanics: Theory and Experiment*, 2005, P02001.

### Examples

```
## generate a random binary tripartite network
set.seed(12)
Net <- build_toy_net(11,15,16,0.2)
pc(Net)

##empirical network
data(PPH_Coltparkmeadow)
Net <- PPH_Coltparkmeadow
pc(Net)
set.seed(13)
library(igraph)
E(Net)$weight<-runif(length(E(Net)),0.1,1)#random weights assigned
pc(Net,weighted=TRUE)

##input as binary matrices,with row names.
md1 <- matrix(sample(c(0,1),8*11,replace=TRUE),8,11)
dimnames(md1) = list(paste0("b",1:8),paste0("c",1:11))
md2 <- matrix(sample(c(0,1),10*12,replace=TRUE),10,12)
dimnames(md2) = list(paste0("b",1:10),paste0("a",1:12))
pc(md1,md2)

##input as weighted matrices,with row numbers as row names.
mdw1 <- matrix(sample(c(rep(0,40),runif(48,0,1))),8,11)
mdw2 <- matrix(sample(c(rep(0,40),runif(80,0,1))),10,12)
pc(mdw1,mdw2)
pc(mdw1,mdw2,weighted=TRUE)
```



---

poc

*Proportion of connector nodes*

---

### Description

Calculating the proportion of connector nodes in the shared set of nodes.

### Usage

```
poc(network.or.subnet_mat1, subnet_mat2 = NULL)
```

### Arguments

`network.or.subnet_mat1`

An "igraph" object or matrix. An "igraph" object with node attribute 'level' or a matrix representing one subnetwork. See details.

`subnet_mat2`

A matrix representing one subnetwork.

### Details

In this package, a tripartite network contains three groups of nodes (a-nodes,b-nodes,c-nodes) and two subnetworks (P includes the links between a-nodes and b-nodes, Q includes the links between b-nodes and c-nodes). Connector nodes belong to b-nodes.

Two types of inputs `network.or.subnet_mat1` can be processed:

- An "igraph" object with node attribute 'level' (0 for a-nodes, 1 for b-nodes,2 for c-nodes).
- Or a matrix representing subnetwork P, and must be input with `subnet_mat2` representing subnetwork Q.

If the inputs are two matrices, please make sure the rows of `network.or.subnet_mat1` and `subnet_mat2` both represent the groups of connector species,i.e, the b-group species. If both matrices have row names, then the function matches row names to produce connector nodes. Otherwise, row numbers are assigned to row names and matched. Within the two matrices (P and Q), columns represents a-group nodes and c-group nodes respectively. Elements in matrices are non-zero values if two nodes are linked with or without weights, and 0 otherwise.

### Value

Return a vector of POC (proportion of connector nodes), number of connector nodes and number of shared species set.

## References

- Battiston, F., Nicosia, V. & Latora, V. (2014) Structural measures for multiplex networks. *Physical Review E*, 89, 032804.
- Dominguez-Garcia, V. and Kefi, S. (2024). The structure and robustness of ecological networks with two interaction types. *PLOS Computational Biology*, 20(1), e1011770.
- Guimera, R. & Amaral, L.A.N. (2005) Cartography of complex networks: modules and universal roles. *Journal of Statistical Mechanics: Theory and Experiment*, 2005, P02001.

## Examples

```
## generate a random binary tripartite network
set.seed(12)
Net <- build_toy_net(11,15,16,0.2)
poc(Net)

## empirical network
data(PPH_Coltparkmeadow)
Net <- PPH_Coltparkmeadow
poc(Net)

##input as binary matrices,with row names.
md1 <- matrix(sample(c(0,1),8*11,replace=TRUE),8,11)
dimnames(md1) = list(paste0("b",1:8),paste0("c",1:11))
md2 <- matrix(sample(c(0,1),10*12,replace=TRUE),10,12)
dimnames(md2) = list(paste0("b",1:10),paste0("a",1:12))
poc(md1,md2)

##input as weighted matrices,with row numbers as row names.
mdw1 <- matrix(sample(c(rep(0,40),runif(48,0,1))),8,11)
mdw2 <- matrix(sample(c(rep(0,40),runif(80,0,1))),10,12)
poc(mdw1,mdw2)
```

---

PPH\_Coltparkmeadow      *An empirical tripartite pollinator-plant-herbivore network*

---

## Description

A pollinator-plant-herbivore tripartite network from Villa-Galaviz et. al. 2020. *Journal of Animal Ecology*

## Usage

```
data(PPH_Coltparkmeadow)
```

**Details**

An 'igraph' object. This is an empirical tripartite network provided in Villa-Galaviz et. al. 2020. This network has three guilds of species: pollinators, plants and herbivores. Pollinators and plants form the subnetwork with mutualistic interactions, and plants and herbivores form the subnetwork with antagonistic interactions. No intra-guild interactions. Plants are the shared set of species.

**References**

Villa-Galaviz, E., S. M. Smart, E. L. Clare, S. E. Ward, and J. Memmott. 2021. Differential effects of fertilisers on pollination and parasitoid interaction networks. *Journal of animal ecology* 90:404-414.

**Examples**

```
data(PPH_Coltparkmeadow)
```

---

role_sim	<i>Similarity of roles of interconnecting species</i>
----------	---

---

**Description**

The average of Similarity of 65 roles of interconnecting species of tripartite network.

**Usage**

```
role_sim(spe_role_mat)
```

**Arguments**

spe\_role\_mat A matrix of 70 columns representing the roles of interconnecting species in the motifs. Columns names are Role1, Role2, Role3 ... Role70.

**Details**

**spe\_role\_mat**

Should acquire from function motif\_role.

**Value**

Return a numeric value.

**Examples**

```
## generate a random tripartite network
set.seed(12)
Net <- build_toy_net(11,15,16,0.2)
role_sim(icmotif_role(Net))

data(PPH_Coltparkmeadow)
Net <- PPH_Coltparkmeadow
role_sim(icmotif_role(Net))
```

---

trigraph\_from\_mat      *Transforming two matrices into an igraph object.*

---

**Description**

Transform two matrices into an igraph object.

**Usage**

```
trigraph_from_mat(mat1, mat2, weighted = FALSE)
```

**Arguments**

mat1	A numeric matrix representing the first subnetwork. Rows should be the shared set of species.
mat2	A numeric matrix representing the second subnetwork. Rows should be the shared set of species.
weighted	Logical. Default to FALSE. If TRUE, a weighted measure is provided.

**Details**

In this package, a tripartite network contains three groups of nodes (a-nodes,b-nodes,c-nodes) and two subnetworks (P includes the links between a-nodes and b-nodes, Q includes the links between b-nodes and c-nodes). Connector nodes belong to b-nodes. Please make sure the rows of mat1 and mat2 both represent the groups of connector species,i.e, the b-group species. If both matrices have row names, then the function matches row names to define connector nodes. Otherwise, row numbers are assigned to row names and matched, which might produce an incorreced network. Within the two matrices (P and Q), columns represents a-group nodes and c-group nodes respectively. Elements in matrices are non-zero values if two nodes are linked with or without weights, and 0 otherwise.

**Value**

Return a network of type "igraph".

**Examples**

```

set.seed(12)
MAT <- build_toy_net(11,22,21,0.2,output_matrices=TRUE)
M <- trigraph_from_mat((MAT[[3]]),MAT[[4]])
M

```

---

tri\_null

*Generates null models for tripartite network*


---

**Description**

A wrapper function to generate different null models for binary and quantitative tripartite networks

**Usage**

```

tri_null(
  trinet,
  null_N = 100,
  null_type = c("sauve", "sub_P", "sub_Q", "both_sub"),
  sub_method
)

```

**Arguments**

trinet	An 'igraph' object that represents a tripartite network.
null_N	The number of null models to be generated. Default to 100.
null_type	Character. Should be one of "sauve", "sub_P", "sub_Q" and "both_sub". See details.
sub_method	The method to shuffle subnetworks. Must be provided when null_type = "sub_P", "sub_Q" or "both_sub". a character specifying the null model algorithm listed on the help page of <code>vegan::commsim</code> . If null_type = 'sauve', it will be ignored.

**Details**

In this package, a tripartite network contains three groups of nodes (a-nodes, b-nodes, c-nodes) and two subnetworks (P includes the links between a-nodes and b-nodes, Q includes the links between b-nodes and c-nodes). Connector nodes belong to b-nodes.

**null\_type**

- "sub\_P", "sub\_Q" and "both\_sub" use the null model algorithms from **vegan::commsim** to shuffle single subnetwork or both of them.
- "sauve" rearranges connector species without changing subnetworks, following [sauve et al.\(2016\)](#).

**Value**

Return a list of null models of a tripartite network.

**References**

Sauve, A. M., Thebault, E., Poccock, M. J., & Fontaine, C. (2016). How plants connect pollination and herbivory networks and their contribution to community stability. *Ecology*, 97(4), 908-917.

**Examples**

```
## generate a random tripartite network
set.seed(12)
Net <- build_toy_net(11,15,16,0.2)

data(PPH_Coltparkmeadow)
Net <- PPH_Coltparkmeadow

set.seed(123)
tri_null_list<-tri_null(Net,null_type="both_sub",sub_method="r00")
set.seed(123)
tri_null_list<-tri_null(Net,null_type="sauve")
```

# Index

## \* datasets

PPH\_Coltparkmeadow, 26

build\_toy\_net, 2

coid, 3

cois, 5

hc, 7

icmotif\_count, 8

icmotif\_role, 10

ig\_ddom, 12

ig\_icmotif\_count, 14

ig\_icmotif\_role, 16

ig\_overlap\_guild, 18

Multi\_motif, 20

node\_icc, 21

pc, 23

poc, 25

PPH\_Coltparkmeadow, 26

role\_sim, 27

tri\_null, 29

trigraph\_from\_mat, 28