

Package ‘netassoc’

May 9, 2026

Type Package

Title Inference of Species Associations from Co-Occurrence Data

Version 0.7.2

Date 2026-01-18

Maintainer Benjamin Blonder <benjamin.blonder@berkeley.edu>

Description Infers species associations from community matrices. Uses local and (optional) regional-scale co-occurrence data by comparing observed partial correlation coefficients between species to those estimated from regional species distributions. Extends Gaussian graphical models to a null modeling framework. Provides interface to a variety of inverse covariance matrix estimation methods.

License GPL-3

Depends igraph, infotheo

Imports corpcor, vegan

NeedsCompilation no

Author Benjamin Blonder [aut, cre],
Naia Morueta-Holme [aut]

Repository CRAN

Date/Publication 2026-01-19 04:10:02 UTC

Contents

netassoc-package	2
make_netassoc_network	2
pairwise_association	5
partial_correlation	5
plot_netassoc_matrix	6
plot_netassoc_network	7

Index	9
--------------	----------

 netassoc-package

Inference of Species Associations from Co-Occurrence Data

Description

Infers species associations from community matrices. Uses local and (optional) regional-scale co-occurrence data by comparing observed partial correlation coefficients between species to those estimated from regional species distributions. Extends Gaussian graphical models to a null modeling framework. Provides interface to a variety of inverse covariance matrix estimation methods.

Author(s)

Benjamin Blonder, Naia Morueta-Holme

Maintainer: Benjamin Blonder <bblonder@gmail.com>

References

Morueta-Holme, N., Blonder, B., et al. A network approach for inferring species associations from co-occurrence data. (in review)

 make_netassoc_network *Infer species-association network*

Description

Infers a species association network by determining which co-occurrence patterns between species are more or less likely than expected under a null model of community assembly. Defaults to estimation of association using a robust shrinkage estimator for inverse covariance matrices.

Usage

```
make_netassoc_network(obs, nul=vegan::permatfull(obs)$perm[[1]],
  method="partial_correlation", args=list(method="shrinkage", verbose=FALSE),
  p.method="fdr", alpha=0.05, numnulls=1000,
  plot=TRUE, plot.legend=TRUE, plot.title=TRUE, verbose=TRUE)
```

Arguments

obs	A m x n community matrix describing the abundance or presence/absence of m species at n sites. Represents the observed data.
nul	A m x n community matrix describing the abundance or presence/absence of m species at n sites. Represents the regional null expectation data. The default value is a resampling of the observed data that preserves row and column sums, but this default method is not recommended.

method	The name of a function used to calculate relationships between species. The function must accept at least the arguments <code>mat</code> , a $m \times n$ (species \times site) matrix. Defaults to partial_correlation .
args	A list of additional arguments to be passed to the method function.
p.method	The method used to correct p-values for multiple comparisons. See p.adjust for options.
alpha	Analysis-wide Type I error rate, controlled via the argument <code>p.method</code> .
numnulls	Number of resamples of the <code>nul</code> matrix used to assemble null communities. Larger values produce more accurate results.
plot	If TRUE, plots all intermediate matrices calculated by the algorithm. Can be used to visualize input and output.
plot.title	If TRUE, adds titles to diagnostic plots.
plot.legend	If TRUE, adds legends to diagnostic plots.
verbose	If TRUE, prints status updates and progress bars during calculations.

Details

Steps taken are:

- 1) obtaining input data and trimming to eliminate species that do not occur in any site
- 2) resampling a set of null community matrices from the expectation with the same richness and abundance as the observed community
- 3) calculating species co-occurrence scores for each pair of species within the observed matrix and all resampled null matrices
- 4) calculating standardized effect sizes and p-values for species' co-occurrence scores
- 5) thresholding effect sizes to retain only significant associations
- 6) converting matrix of scores to association network

The resulting network can be analyzed using functions from the `igraph` network package.

The user should specify a `nul` matrix of the same dimensionality as `obs` based on some regional distribution modeling approach (e.g. MaxEnt). The default reshuffling method is not recommended but provided to allow immediate output from the function.

This process by default builds a Gaussian graphical model via estimating an inverse covariance matrix (precision matrix, which can be used to calculate partial correlation coefficients) for all species pairs. This graph is then compared to a distribution of null graphs, such that the final output is a graph with edge weights corresponding to standardized effect sizes after correction for multiple comparisons.

A range of different methods are provided in [partial_correlation](#) for estimating relationships between species. Note that while a method is provided for the graphical lasso (L1-regularization) its use is not recommended, as it will produce very sparse null networks and then a narrow (or singular) distribution of null edge weights.

The inverse covariance methods implemented in [partial_correlation](#) result in symmetric association metrics. Non-symmetric metrics (e.g. describing predation or commensalism) are possible mathematically but their usage is not well-established. For an example of how to implement these, see [pairwise_association](#).

Value

A list with the following components:

matrix_spsite_obs	Trimmed obs matrix
matrix_spsite_nul	Trimmed nul matrix
matrix_spsp_obs	Observed co-occurrence scores for all species
matrix_spsp_ses_thresholded	Observed co-occurrence scores for all species after removing those with non-significant p-values
matrix_spsp_pvalue	P-values for all species after correction for multiple comparisons
network_all	An igraph object representing the association network
network_pos	An igraph object representing an association network including only positive associations
network_neg	An igraph object representing an association network including only negative associations

See Also

vegan::permat

Examples

```

set.seed(1)
nsp <- 10
nsi <- 50
m_obs <- floor(matrix(rpois(nsp*nsi,lambda=5),ncol=nsi,nrow=nsp))
m_nul <- floor(matrix(rpois(nsp*nsi,lambda=5),ncol=nsi,nrow=nsp))

m_obs[1,1:(nsi/2)] <- rpois(n=nsi/2,lambda=20)
m_obs[2,1:(nsi/2)] <- rpois(n=nsi/2,lambda=20)

n <- make_netassoc_network(m_obs, m_nul,
  method="partial_correlation",args=list(method="shrinkage"),
  p.method='fdr',
  numnulls=100, plot=TRUE,alpha=0.05)

# experimental demonstration of non-symmetric metrics
#n <- make_netassoc_network(m_obs, m_nul,
# method="pairwise_association",args=list(method="condentropy"),
# p.method='fdr',
# numnulls=100, plot=TRUE,alpha=0.05)

n$network_all

```

pairwise_association *Pairwise associations*

Description

Computes pairwise associations between every row (species) in a species x site matrix. Note that usage of this function is advantageous when non-symmetric association metrics are desired, but the pairwise computation will prevent accounting for indirect effects between species. As such this function should be considered preliminary, and its use experimental.

Usage

```
pairwise_association(mat, method = "condentropy")
```

Arguments

mat	A m x n (species x site) matrix
method	The name of a function to call to calculate an association score. Must take two vector arguments (X,Y) and return a single numeric value. Default argument uses conditional information entropy statistic, although other functions (e.g. Jaccard similarity) are possible.

Value

A n x n (species x species) matrix with NA diagonal values. May be non-symmetric depending on the method used.

Examples

```
nsp <- 10
nsi <- 50
m_obs <- floor(matrix(rpois(nsp*nsi,lambda=5),ncol=nsi,nrow=nsp))
m_obs[1,1:(nsi/2)] <- rpois(n=nsi/2,lambda=20)

spxsp <- pairwise_association(m_obs, method="condentropy")
image(spxsp)
```

partial_correlation *Partial correlation coefficients*

Description

Estimates the inverse covariance matrix then uses this matrix to calculate partial correlation coefficients.

Assumes that matrix rows correspond to different variables of interest.

The one exception is if method="correlation"; see below for details.

Usage

```
partial_correlation(mat, method, verbose=FALSE)
```

Arguments

mat	Input matrix.
method	One of the following: "shrinkage": Shrinkage estimate. The shrinkage parameter is estimated via an analytic formula. Uses corpcor::invcov.shrink. "exact": Directly calculates inverse covariance matrix. Results may be sensitive to highly collinear inputs. Use not recommended. Uses base::solve. "correlation": Directly calculates correlation matrix. Does not return partial correlation coefficients as in all other cases. Use not recommended. Uses stats::cor.
verbose	Binary flag determining whether diagnostic output is shown.

Value

Returns a $m \times m$ upper triangular matrix of partial correlation coefficients from an input $m \times n$ matrix.

Examples

```
# load highly collinear economic data time series
data(longley)
longley_ss <- t(longley[,c(1:5,7)]) # put data in correct input format

colors <- colorRampPalette(c("red","white","blue"))(10)
pc_shrinkage <- partial_correlation(longley_ss,method="shrinkage")

image(pc_shrinkage,zlim=c(-1,1),col=colors)
```

plot_netassoc_matrix *Plots matrix with colormap*

Description

Plots species x species or species x site matrix with color map

Usage

```
plot_netassoc_matrix(data,
  colors, onesided=FALSE, main="",
  legend=TRUE, axis=TRUE, title=TRUE, cex.axis=0.5)
```

Arguments

data	Input matrix; assumed to have dimension names
colors	Vector of colors
onesided	If TRUE, assumes that colors maps to an entirely positive or entirely negative scale, i.e. without zero-crossings.
main	Title of plot.
legend	If TRUE, plots a colorbar legend.
axis	If TRUE, labels axes with dimension names.
title	If TRUE, plots a title.
cex.axis	Expansion factor for axis labels.

Value

None; used for the side effect of making a plot.

Examples

```
nsp <- 10
nsites <- 30
obs <- matrix(rpois(n=nsp*nsites,10),
  nrow=nsp,ncol=nsites,
  dimnames=list(paste("Species",1:nsp),paste("Site",1:nsites)))
plot_netassoc_matrix(obs, onesided=TRUE, col=heat.colors(5))

int <- matrix(rnorm(n=nsp^2),
  nrow=nsp,ncol=nsp,
  dimnames=list(paste("Species",1:nsp),paste("Species",1:nsp)))
plot_netassoc_matrix(int, onesided=FALSE,
  col=colorRampPalette(c("red","white","blue"))(50))
```

plot_netassoc_network *Plots species association network*

Description

Draws a network of species associations. By default edge widths are proportional to association strength and edge color reflects association type (blue, positive; red, negative).

Usage

```
plot_netassoc_network(network, layout = layout_nicely(network),
  vertex.label = V(network)$name,
  vertex.color = NA,
  vertex.shape = "none",
  vertex.label.color = "black",
  vertex.label.family = "sans",
```

```

edge.width = NULL,
edge.color = NULL,
edge.arrow.size = 0.2,
vertex.label.cex = 0.5,
legend = TRUE,
...)
```

Arguments

network	An igraph object corresponding to the association network
layout	Graphical layout. See <code>igraph::layout</code> .
vertex.label	String labels for species.
edge.width	Edge widths for links between species.
edge.color	Edge colors for links between species.
vertex.color	Vertex colors for species.
vertex.label.color	Vertex label colors for species.
vertex.shape	Vertex shape for species.
edge.arrow.size	Edge arrow size for links between species.
vertex.label.cex	Vertex label expansion factor for species.
vertex.label.family	Vertex shape font family for species.
legend	If TRUE, plots a scale legend.
...	Other arguments to be passed to <code>plot.igraph</code> .

Examples

```

# generate random data
set.seed(5)
nsp <- 10
nsi <- 5
m_obs <- floor(matrix(rgamma(nsp*nsi,shape=5),ncol=nsi,nrow=nsp))
m_nul <- floor(matrix(rexp(nsp*nsi,rate=0.05),ncol=nsi,nrow=nsp))

n <- make_netassoc_network(m_obs, m_nul, numnulls=100, plot=TRUE,alpha=0.5)

# plot
plot_netassoc_network(n$network_all)

# plot using circular layout
plot_netassoc_network(n$network_all, layout=layout.circle(n$network_all))
```

Index

* **package**

netassoc-package, [2](#)

make_netassoc_network, [2](#)

netassoc (netassoc-package), [2](#)

netassoc-package, [2](#)

p.adjust, [3](#)

pairwise_association, [3](#), [5](#)

partial_correlation, [3](#), [5](#)

plot.igraph, [8](#)

plot_netassoc_matrix, [6](#)

plot_netassoc_network, [7](#)