

Package: **dissCqN** (via r-universe)

August 26, 2024

Type Package

Title Multiple Assemblage Dissimilarity for Orders $q = 0-N$

Version 0.1.0

Description Calculate multiple or pairwise dissimilarity for orders $q = 0-N$ (CqN; Chao et al., 2008 <[doi:10/fcvm63](https://doi.org/10/fcvm63)>) for a set of species assemblages or interaction networks.

URL <https://murphymv.github.io/dissCqN/>,
<https://github.com/murphymv/dissCqN>

BugReports <https://github.com/murphymv/dissCqN/issues>

Depends R ($\geq 4.0.0$)

Imports parallel, stats

Suggests knitr, markdown, rmarkdown, SpadeR, vegan

VignetteBuilder knitr

License GPL (≥ 3)

Encoding UTF-8

LazyData true

LazyDataCompression xz

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.2

Language en-GB

Repository <https://murphymv.r-universe.dev>

RemoteUrl <https://github.com/murphymv/disscqN>

RemoteRef HEAD

RemoteSha 0ad48acd449106a15cf8c09bd250a29e89d14363

Contents

dissCqN	2
intMat	4
netShared	5

dissCqN *Multiple Assemblage Dissimilarity*

Description

Multiple assemblage dissimilarity for orders $q = 0-N$.

Usage

```
dissCqN(
  mat,
  q = 0:2,
  pairwise = FALSE,
  compare.sub = NULL,
  shared.spp = FALSE,
  parallel = "no",
  ncpus = NULL,
  cl = NULL
)
```

Arguments

mat	A matrix with assemblages in rows and species or species interactions in columns. Alternatively, a list of matrices, which will be interpreted as interaction networks and used to construct an assemblage x interaction matrix.
q	Integer, the order(s) of q for which to calculate dissimilarity. Can be any set of integers between 0 and N (the number of assemblages in mat).
pairwise	Logical, whether to calculate pairwise, rather than multiple assemblage, dissimilarity.
compare.sub	Subsets of assemblages to compare pairwise. These should be supplied as a list of two sets of assemblage names or indices. If only one set is supplied, this is compared to all other assemblages in mat. If more than two sets are supplied, only the first two are used. If NULL (default), all assemblages are compared.
shared.spp	Logical, whether to compare networks of shared species only (if mat is a list of networks).
parallel	The type of parallel processing to use, if any. Can be one of "snow", "multicore", or "no" (for none – the default). Passed to <code>pSapply()</code> .
ncpus	Number of system cores to use for parallel processing. If NULL (default), all available cores are used.
cl	Optional cluster to use if parallel = "snow". If NULL (default), a local cluster is created using the specified number of cores.

Details

Dissimilarity is calculated here for multiple species assemblages (or interaction networks) via the CqN generalisation of similarity indices (Chao et al., 2008; Jost et al., 2011). Increasing the value of q increases the 'depth' of the measure, that is, how much emphasis is placed on changes in relative abundance of the most common species. Setting $q = 0$ represents the qualitative Sørensen index (Sørensen, 1948), where rare and common species are treated equally. $q > 0$ is more sensitive to common species, with $q = 1$ representing the Shannon-based Horn index (Horn, 1966) and $q = 2$ the Simpson-based Morisita-Horn index (Horn, 1966; Morisita, 1959). For $N > 2$, indices are generalised to consider species shared across multiple assemblages (Diserud & Ødegaard, 2007; eqns. 6.3-6.5 in Jost et al., 2011). For $q \geq 2 \leq N$, common species increasingly dominate the measure, and it can then be interpreted as the ratio of two probabilities of randomly sampling q individuals of the same species from the N assemblages, where 1) the individuals came from at least one different assemblage (qG_D) and 2) they all came from the same assemblage (qG_S) (Jost et al., 2011). Dissimilarity is thus:

$$1 - {}^qG_D / {}^qG_S$$

Pairwise dissimilarity can be calculated for all or a subset of the assemblages (or networks) in `mat`, in which case a dissimilarity matrix is returned (one for each value of q). If comparing subsets, the names or indices of assemblages to compare should be supplied to `compare.sub`. Note that pairwise calculation may take a long time if N is large, in which case parallel processing may speed up results (e.g. `parallel = "snow"`).

If `shared.spp = TRUE` and `mat` is a list of interaction networks (as matrices), multiple or pairwise interaction dissimilarity will be calculated for networks of shared species only (see `netShared()`). This can be useful to help partition the different components of network dissimilarity, e.g. dissimilarity due to interaction 'rewiring' among shared species vs. that due to species turnover (Poisot et al., 2012).

Value

A numeric vector of dissimilarities, or a pairwise dissimilarity matrix (or list of matrices), for the orders of q .

References

- Chao, A., Jost, L., Chiang, S. C., Jiang, Y.-H., & Chazdon, R. L. (2008). A Two-Stage Probabilistic Approach to Multiple-Community Similarity Indices. *Biometrics*, 64(4), 1178–1186. doi:10/fcvm63
- Diserud, O. H., & Ødegaard, F. (2007). A multiple-site similarity measure. *Biology Letters*, 3(1), 20–22. doi:10/bwhfx6
- Horn, H. S. (1966). Measurement of "Overlap" in Comparative Ecological Studies. *The American Naturalist*, 100(914), 419–424. doi:10/b62ct5
- Jost, L., Chao, A., & Chazdon, R. L. (2011). Compositional similarity and beta diversity. In A. E. Magurran & B. J. McGill (Eds.), *Biological Diversity: Frontiers in Measurement and Assessment* (pp. 66–84). Oxford University Press.
- Morisita, M. (1959). Measuring of interspecific association and similarity between communities. *Memoirs of the Faculty of Science, Kyushu Univ., Series E (Biology)*, 3, 65–80.

Poisot, T., Canard, E., Mouillot, D., Mouquet, N., & Gravel, D. (2012). The dissimilarity of species interaction networks. *Ecology Letters*, *15*(12), 1353–1361. doi:10/f4dv37

Sørensen, T. (1948). A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons. *Kongelige Danske Videnskabernes Selskabs Biologiske Skrifter*, *5*, 1–34.

Examples

```
# Sample community data from SpadeR package (three assemblages, 120 species)
data(SimilarityMultData, package = "SpadeR")
d <- SimilarityMultData$Abu

# Multiple-assemblage dissimilarity for q = 0:2
(CqN <- dissCqN::dissCqN(t(d)))

# Compare to empirical CqN values from SpadeR::SimilarityMult()
sim <- SpadeR::SimilarityMult(d, datatype = "abundance", nboot = 1)
CqN_2 <- 1 - c(
  "C0N" = sim$Empirical_richness["C0N(q=0,Sorensen)", "Estimate"],
  "C1N" = sim$Empirical_relative["C1N=U1N(q=1,Horn)", "Estimate"],
  "C2N" = sim$Empirical_relative["C2N(q=2,Morisita)", "Estimate"]
)
stopifnot(all.equal(CqN, CqN_2))

# Pairwise dissimilarity matrices
dissCqN::dissCqN(t(d), pairwise = TRUE)
```

intMat

Assemblage x Species Interaction Matrix

Description

Generate a matrix of assemblages x species interactions from a set of networks.

Usage

```
intMat(net, shared.spp = FALSE, ...)
```

Arguments

`net` An interaction network, or list of networks, supplied as matrices.
`shared.spp` Logical, whether to use networks of shared species only.
`...` Arguments to `netShared()` (if `shared.spp = TRUE`).

Value

A matrix with assemblages in rows and species interactions in columns.

`netShared`*Networks of Shared Species*

Description

Generate interaction networks comprising only the shared species across two or more networks.

Usage

```
netShared(net, pairwise = TRUE, compare.sub = NULL)
```

Arguments

<code>net</code>	A list of two or more networks to compare, supplied as matrices.
<code>pairwise</code>	Logical, whether to compare networks pairwise (default), rather than considering species shared across multiple networks.
<code>compare.sub</code>	Subsets of networks to compare pairwise. These should be supplied as a list of two sets of network names or indices. If only one set is supplied, this is compared to all other networks in <code>net</code> . If more than two sets are supplied, only the first two are used.

Value

A list of networks of shared species. If comparing pairwise, this will be of length $n1 * n2 * 2$ (with $n1$ and $n2$ being the numbers of networks in each set), or if considering multiple networks, the length of the original list.

Note

If comparing networks pairwise, and subsets are not specified, the output will contain network self-comparisons (redundancy).

Index

`dissCqN`, 2

`intMat`, 4

`netShared`, 5

`netShared()`, 3, 4

`pSapply()`, 2