Package 'NST'

September 16, 2020

Type Package

Title Normalized Stochasticity Ratio

Version 3.0.3

Date 2020-09-09

Author Daliang Ning

Maintainer Daliang Ning <ningdaliang@ou.edu>

Imports vegan,parallel,permute,ape,bigmemory,iCAMP

Depends R (>= 3.1.0)

Description To estimate ecological stochasticity in community assembly. Understanding the community assembly mechanisms controlling biodiversity patterns is a central issue in ecology. Although it is generally accepted that both deterministic and stochastic processes play important roles in community assembly, quantifying their relative importance is challenging. The new index, normalized stochasticity ratio (NST), is to estimate ecological stochasticity, i.e. relative importance of stochastic processes, in community assembly. With functions in this package, NST can be calculated based on different similarity metrics and/or different null model algorithms, as well as some previous indexes, e.g. previous Stochasticity Ratio (ST), Standard Effect Size (SES), modified Raup-Crick metrics (RC). Functions for permutational test and bootstrapping analysis are also included. Previous ST is published by Zhou et al (2014) <doi:10.1073/pnas.1324044111>. NST is modified from ST by considering two alternative situations and normalizing the index to range from 0 to 1 (Ning et al 2019) <doi:10.1073/pnas.1904623116>. A modified version, MST, is a special case of NST, used in some recent or upcoming publications, e.g. Liang et al (2019) <doi:10.1101/638908>. SES is calculated as described in Kraft et al (2011) <doi:10.1126/science.1208584>. RC is calculated as reported by Chase et al (2011) <doi:10.1890/es10-00117.1> and Stegen et al (2013) <doi:10.1038/ismej.2013.93>. Version 3 added NST based on phylogenetic beta diversity, used by Ning et al (2020) <doi:10.1101/2020.02.22.960872>.

License GPL-2

NeedsCompilation no

Repository CRAN

Date/Publication 2020-09-16 05:30:02 UTC

R topics documented:

`NST-package` *Normalized Stochasticity Ratio*

Description

This package is to estimate ecological stochasticity in community assembly based on beta diversity. Various indexes can be calculated, including Stochasticity Ratio (ST), Normalized Stochasticity Ratio (NST), Modified Stochasticity Ratio (MST), Standard Effect Size (SES), and modified Raup-Crick metrics (RC), based on various taxonomic and phylogenetic dissimilarity metrics and different null model algorithms.

Version 2.0.4: Update citation and references. Emphasize that NST variation should be calculated from nst.boot rather than pairwise NST.ij from tNST. Emphasize that different group setting in tNST may lead to different NST results. Version 3.0.1: Add NST based on phylogenetic beta diversity (pNST). Version 3.0.2: debug pNST. Version 3.0.3: remove setwd in functions; change dontrun to donttest and revise save.wd in help doc.

Details

Package:	NST
Type:	Package
Version:	3.0.3
Date:	2020-09-09
License:	GPL-2

Author(s)

Daliang Ning <ningdaliang@ou.edu>

References

Ning D., Deng Y., Tiedje J.M. & Zhou J. (2019) A general framework for quantitatively assessing ecological stochasticity. Proceedings of the National Academy of Sciences 116, 16892-16898. doi:10.1073/pnas.1904623116.

Zhou J, Deng Y, Zhang P, Xue K, Liang Y, Van Nostrand JD, Yang Y, He Z, Wu L, Stahl DA, Hazen TC, Tiedje JM, and Arkin AP. (2014) Stochasticity, succession, and environmental perturbations in a fluidic ecosystem. Proceedings of the National Academy of Sciences of the United States of America 111, E836-E845. doi:10.1073/pnas.1324044111.

Examples

```
data(tda)
comm=tda$comm
group=tda$group
tnst=tNST(comm=comm, group=group, dist.method="jaccard",
          abundance.weighted=TRUE, rand=100,
          nworker=1, null.model="PF", between.group=TRUE,
          SES=TRUE, RC=TRUE)
```

---

ab.assign *Randomly draw individuals into species according to specified probabilities*

Description

This funciton is to assign abundances to species when randomizing communities based on null models considering abundances. Individuals are randomly drawn into species according to the specified probabilities.

Usage

```
ab.assign(comm.b, samp.ab, prob.ab)
```

Arguments

comm.b	numeric matrix, binary (present/absent) community data, rownames are sample/site names, colnames are species names.
samp.ab	numeric vector, total abundances (total individual numbers) in each sample.
prob.ab	numeric matrix, probability of each species into which the individuals in a certain sample are drawn.

Details

This function is called by the function taxo.null to generate randomized communities.

Value

A matrix of community data with abundances is returned. rownames are sample/site names, and colnames are species names.

Note

version 1.0: 2015.10.22

Author(s)

Daliang Ning

References

Stegen JC, Lin X, Fredrickson JK, Chen X, Kennedy DW, Murray CJ, Rockhold ML, and Konopka A. Quantifying community assembly processes and identifying features that impose them. Isme Journal 7, 2069-2079 (2013).

See Also

taxo.null

Examples

```
data(tda)
comm=tda$comm
comm.b=comm
comm.b[comm.b>0]=1
samp.ab=rowSums(comm)
prob.ab=matrix(colSums(comm),nrow=nrow(comm),ncol=ncol(comm),byrow=TRUE)
comm.rand=ab.assign(comm.b,samp.ab,prob.ab)
```

---

beta.g *Various taxonomic beta diversity indexes*

Description

This function can simultaneously calculate various taxonomic dissimilarity indexes, mainly based on vegdist from package vegan.

Usage

```
beta.g(comm, dist.method="bray", abundance.weighted=TRUE,
       as.3col=FALSE,out.list=TRUE)
chaosorensen(comm, dissimilarity=TRUE, to.dist=TRUE)
chaojaccard(comm, dissimilarity=TRUE, to.dist=TRUE)
```

Arguments

comm	Community data matrix. rownames are sample names. colnames are species names.
dist.method	A character or vector indicating one or more index(es). match to "manhattan", "euclidean", "canberra", "bray", "kulczynski", "jaccard", "gower", "altGower", "morisita", "horn", "mountford", "raup" , "binomial", "chao", "cao", "mahalanobis", "mGower", "mEuclidean", "mManhattan", "chao.jaccard", "chao.sorensen". default is "bray"
abundance.weighted	Logic, consider abundances or not (just presence/absence). default is TRUE.
as.3col	Logic, output a 3-column matrix (TRUE) or a square matrix (FALSE) for each index. default is FALSE.
out.list	Logic, if using multiple indexes, output their results as a list (TRUE) or a matrix combining all 3-column matrixes (FALSE). if out.list=FALSE, as.3col will be forced to be TRUE. default is TRUE.
dissimilarity	Logic, calculate dissimilarity or similarity. default is TRUE, means to return dissimilarity.
to.dist	Logic, return distance object or squared matrix. default is TRUE, means to return distance object.

Details

All the taxonomic beta diversity indexes are mainly calculated by vegdist in package vegan, except following methods:

mGower, mEuclidean, and mManhattan are modified from Gower, Euclidean, and Manhattan, respectively, according to the method reported previously (Anderson et al 2006).

chao.jaccard and chao.sorensen are calculated as described previously (Chao et al 2005), using open-source code from R package "fossil" (Vavrek 2011), but output as dissimilarity for each pairwise comparison.

Value

beta.g will return a square matrix of each index if as.3col=FALSE, and combined as a list if out.list=TRUE (default). A 3-column matrix with first 2 columns indicating the pairwised samples will be output for each index if as.3col=TRUE, and combined as a list if out.list=TRUE or integrated into one matrix if out.list=FALSE.

chaosorensen and chaojaccard will return a distance object (if to.dist=TRUE) or a squared matrix (if to.dist=FALSE).

Note

Version 2: 2019.5.10 Version 1: 2015.9.25

Author(s)

Daliang Ning

References

Jari Oksanen, F. Guillaume Blanchet, Michael Friendly, Roeland Kindt, Pierre Legendre, Dan McGlinn, Peter R. Minchin, R. B. O'Hara, Gavin L. Simpson, Peter Solymos, M. Henry H. Stevens, Eduard Szoecs and Helene Wagner (2019). vegan: Community Ecology Package. R package version 2.5-4.

Anderson MJ, Ellingsen KE, & McArdle BH (2006) Multivariate dispersion as a measure of beta diversity. Ecol Lett 9(6):683-693.

Chao, A., R. L. Chazdon, et al. 2005. A new statistical approach for assessing similarity of speciescomposition with incidence and abundance data. Ecology Letters 8: 148-159

Vavrek, Matthew J. 2011. fossil: palaeoecological and palaeogeographical analysis tools. Palaeontologia Electronica, 14:1T.

Others cited in the help document of vegdist in R package vegan.

See Also

tNST

Examples

```
data(tda)
comm=tda$comm
# calculate one index
beta.bray=beta.g(comm=comm,as.3col=TRUE)

# calculate multiple indexes
beta.td=beta.g(comm=comm,dist.method=c("bray","jaccard","euclidean",
               "manhattan","binomial","chao","cao"),
               abundance.weighted = TRUE,out.list=FALSE)
```

---

beta.limit *Upper limit of different beta diversity (dissimilarity) indexes*

Description

Upper limit value of each abundance-based or incidence-based dissimilarity index.

Usage

```
data("beta.limit")
```

Format

A data frame with 18 observations on the following 2 variables.

`Dmax.in` numeric, upper limit of incidence-based dissimilarity

`Dmax.ab` numeric, upper limit of abundance-based dissimilarity

Examples

```
data(beta.limit)
```

`bmntd.big` *beta mean nearest taxon distance (betaMNTD) from big data*

Description

Calculates beta MNTD (beta mean nearest taxon distance, Webb et al 2008) for taxa in each pair of communities in a givern community matrix, using bigmemory (Kane et al 2013) to deal with too large dataset.

Usage

```
bmntd.big(comm, pd.desc = "pd.desc", pd.spname, pd.wd,
          spname.check = FALSE, abundance.weighted = TRUE,
          exclude.conspecifics = FALSE, time.output = FALSE)
```

Arguments

`comm`	matrix or data.frame, community data matrix, rownames are sample names, colnames are taxa ids.
`pd.desc`	character, the name to describe bigmemory file of phylogenetic distance matrix, default is "pd.desc".
`pd.spname`	vector, the OTU ids (species names) in exactly the same order as the phylogenetic matrix rows or columns
`pd.wd`	the path of the folder saving the phylogenetic distance matrix.
`spname.check`	logic, whether to check the OTU ids (species names) in community matrix and phylogenetic distance matrix are the same.
`abundance.weighted`	logic, whether weighted by species abundance, default is TRUE, means weighted.
`exclude.conspecifics`	logic, whether conspecific taxa in different communities be exclude from beta MNTD calculations, default is FALSE.
`time.output`	logic, whether to count calculation time, default is FALSE.

Details

beta mean nearest taxon distance for taxa in each pair of communities. Improved from 'comdistnt' in package 'picante'(Kembel et al 2010). This function adds bigmemory part (Kane et al 2013) to deal with large dataset.

Value

result is a distance object.

Note

Version 3: 2020.9.9, remove setwd; change dontrun to donttest and revise save.wd in help doc. Version 2: 2020.8.22, add to NST package, update help document. Version 1: 2017.3.13

Author(s)

Daliang Ning (ningdaliang@ou.edu)

References

Webb, C.O., Ackerly, D.D. & Kembel, S.W. (2008). Phylocom: software for the analysis of phylogenetic community structure and trait evolution. Bioinformatics, 24, 2098-2100.

Kembel, S.W., Cowan, P.D., Helmus, M.R., Cornwell, W.K., Morlon, H., Ackerly, D.D. et al. (2010). Picante: R tools for integrating phylogenies and ecology. Bioinformatics, 26, 1463-1464.

Kane, M.J., Emerson, J., Weston, S. (2013). Scalable Strategies for Computing with Massive Data. Journal of Statistical Software, 55(14), 1-19. URL http://www.jstatsoft.org/v55/i14/.

Examples

```
data("tda")
comm=tda$comm
tree=tda$tree
# since it needs to save some file to a certain folder,
# the following code is set as 'not test'.
# but you may test the code on your computer
# after change the folder path for 'save.wd'.

  save.wd=tempdir() # please change to the folder you want to use.
  nworker=2 # parallel computing thread number
  pd.big=iCAMP::pdist.big(tree = tree,wd = save.wd, nworker = nworker)
  bmntd.wt=bmntd.big(comm=comm, pd.desc = pd.big$pd.file,
                     pd.spname = pd.big$tip.label, pd.wd = pd.big$pd.wd,
                     abundance.weighted = TRUE)
```

`dist.3col` *Transform distance matrix to 3-column matrix*

Description

Transform a distance matrix to a 3-column matrix in which the first 2 columns indicate the pairwised samples/species names.

Usage

```
dist.3col(dist)
```

Arguments

dist	a square matrix or distance object with column names and row names.

Details

In many cases, a 3-column matrix is easier to use than a distance matrix.

Value

name1	1st column, the first item of pairwised two items
name2	2nd column, the second item of pairwised two items
dis	3rd column, distance value of the pairwised two itmes

Note

Version 1: 2015.5.17

Author(s)

Daliang Ning

Examples

```
data(tda)
comm=tda$comm
bray=beta.g(comm,dist.method="bray")
bray.3col=dist.3col(bray)
```

---

match.name *Check and ensure the consistency of IDs in different objects.*

Description

This function is usually used to check the consistency of species or samples names in different data table (e.g. OTU table and phylogenetic distance matrix). it can be used to check row names and/or column names of different matrixes, names in vector(s) or list(s), and tip.lable in tree(s)

Usage

```
match.name(name.check=integer(0), rn.list=list(integer(0)),
           cn.list=list(integer(0)), both.list=list(integer(0)),
           v.list=list(integer(0)), lf.list=list(integer(0)),
           tree.list=list(integer(0)), group=integer(0),
           rerank=TRUE, silent=FALSE)
```

Arguments

name.check	A character vector, indicating reference name list or the names you would like to keep. If not available, a union of all names is set as reference name list.
rn.list	A list object, including the matrix(es) of which the row names will be check. rn.list must be set in a format like "rn.list=list(A=A,B=B)". default is nothing.
cn.list	A list object, including the matrix(es) of which the column names will be check. cn.list must be set in a format like "cn.list=list(A=A,B=B)". default is nothing.
both.list	A list object, including the matrix(es) of which both column and row names will be check. both.list must be set in a format like "both.list=list(A=A,B=B)". default is nothing.
v.list	A list object, including the vector(s) of which the names will be check. v.list must be set in a format like "v.list=list(A=A,B=B)".default is nothing.
lf.list	A list object, including the list(s) of which the names will be check. lf.list must be set in a format like "lf.list=list(A=A,B=B)".default is nothing.
tree.list	A list object, including the tree(s) of which the tip.label names will be check. tree.list must be set in a format like "tree.list=list(A=A,B=B)".default is nothing.
group	a vector or one-column matrix/data.frame indicating the grouping information of samples or species, of which the sample/species names will be check.
rerank	Logic, make all names in the same rank or not. Default is TRUE
silent	Logic, whether to show messages. Default is FALSE, thus all messages will be showed.

Details

In many cases and functions, species names and samples names must be checked and set in the same rank. Sometimes, we also need to select some samples or species as necessary. This function can help.

Value

Return a list object, new matrixes with the same row/column names in the same rank. Some messages will return if some names are removed or all names matches very well.

Note

Version 3: 2017.3.13 Version 2: 2015.9.25

Author(s)

Daliang Ning

Examples

```
data(tda)
comm=tda$comm
group=tda$group
# check sample IDs
```

```
sampc=match.name(rn.list=list(com=comm,grp=group))
# output comm and group with consitent IDs.
comc=sampc$com
grpc=sampc$grp
```

`nst.boot` *Bootstrapping test for ST and NST*

Description

To test the distribution of ST and NST in each group, and the significance of ST and NST difference between each pair of groups.

Usage

```
nst.boot(nst.result, group=NULL, rand=999, trace=TRUE,
         two.tail=FALSE, out.detail=FALSE, between.group=FALSE,
         nworker=1)
```

Arguments

`nst.result`	list object, the output of tNST, must have "details", thus the output.rand must be TRUE in tNST function.
`group`	n x 1 matrix, if the grouping is different from the nst.result. default is NULL, means to use the grouping in nst.result.
`rand`	integer, random draw times for bootstrapping test.
`trace`	logic, whether to show message when randomizing.
`two.tail`	logic, the p value is two-tail or one-tail.
`out.detail`	logic, whether to output details rather than just summarized results.
`between.group`	Logic, whether to calculate for between-group turnovers. default is FALSE.
`nworker`	for parallel computing. Either a character vector of host names on which to run the worker copies of R, or a positive integer (in which case that number of copies is run on localhost). default is 1, means not to use parallel computing.

Details

Normalized stochasticity ratio (NST, Ning et al 2019) is a index to estimate average stochasticity within a group of samples. Bootstrapping is an excellent method to evaluate the statistical variation. Since the observed/null dissimilarity values are not independent (pairwise comparisons), bootstrapping should be random draw of samples rather than the pairwise values. Bootstrapping for stochasticity ratio (ST, Zhou et al 2014) is also performed.

Value

Output is a list object, includes

`ST.summary` obs, the ST value of observed samples; mean, mean value of bootstrapping results; stdev, standard deviation; Min, minimal value; Quantile25, quantile of 25 percent; Median, median value; Quantile75, 75 percent quantile; Max, maximum value; LowerWhisker, LowerHinge, Median, HigherHinge, HigherWhisker, values for box-and-whisker plot; Outliers, outliers in bootstrapping values which out of the range of 1.5 fold of IQR.

`NST.summary` Same as above, but for NST values

`ST.compare` Comparison between each pair of groups, and p values. p.wtest, p value from wilcox.test; w.value, w value from wilcom.test; p.count, p value calculated by directly comparing all values in two groups; ..noOut, means outliers were not included for significance test. In principle, p.count or p.count.noOutis preferred, and others have defects.

`NST.compare` Same as above, but for NST values

`ST.between.summary`
Similar to ST.summary, but for between-group comparisons.

`NST.between.summary`
Similar to NST.summary, but for between-group comparisons.

`detail` a list object. ST.boot, a list of bootstrapping detail results of ST for each group, each element in the list means the result of one random draw; NST.boot, bootstrapping results of NST; ST.boot.rmout, bootstrapping results of ST without outliers; NST.boot.rmout, bootstrapping results of NST without outliers; STb.boot, NSTb.boot, STb.boot.rmout, and NSTb.boot.rmout have the same meanning but for between-group comparisons.

Note

Version 3: 2019.10.8. Update reference. Version 2: 2019.5.10 Version 1: 2018.1.9

Author(s)

Daliang Ning

References

Ning D., Deng Y., Tiedje J.M. & Zhou J. (2019) A general framework for quantitatively assessing ecological stochasticity. Proceedings of the National Academy of Sciences 116, 16892-16898. doi:10.1073/pnas.1904623116.

See Also

`tNST`, `nst.panova`

Examples

```
data(tda)
comm=tda$comm
group=tda$group
tnst=tNST(comm=comm, group=group, rand=20,
          output.rand=TRUE, nworker=1)
# rand is usually set as 1000, here set rand=20 to save test time.

nst.bt=nst.boot(nst.result=tnst, group=NULL, rand=99,
                trace=TRUE, two.tail=FALSE, out.detail=FALSE,
                between.group=FALSE, nworker=1)
# rand is usually set as 999, here set rand=99 to save test time.
```

---

nst.panova *Permutational multivariate ANOVA test for ST and NST*

Description

Permutational multivariate ANOVA test for stochasticity ratio and normalized stochasticity ratio between treatments

Usage

```
nst.panova(nst.result, group=NULL, rand=999, trace=TRUE)
```

Arguments

nst.result	list object, the output of nsto, must have "details"
group	nx1 matrix, if the grouping is different from the nst.result. default is NULL, means to use the grouping in nst.result.
rand	integer, randomization times for permuational test
trace	logic, whether to show message when randomizing.

Details

coming later

Value

Output is a data.frame object.

index	name of index
group1	treatment/group name
group2	treatment/group name
Index.group1	index value in group1

Index.group2	index value in group2
Difference	index.group1 - index.group2
F.obs	F value
P.anova	P value of parametric ANOVA test
P.panova	P value of permutational ANOVA test
P.perm	P value of permutational test of the difference

Note

Version 3: 2019.10.8. Update reference. Version 2: 2019.5.10 Version 1: 2017.12.30

Author(s)

Daliang Ning

References

Ning D., Deng Y., Tiedje J.M. & Zhou J. (2019) A general framework for quantitatively assessing ecological stochasticity. Proceedings of the National Academy of Sciences 116, 16892-16898. doi:10.1073/pnas.1904623116.

See Also

tNST, nst.boot

Examples

```
data(tda)
comm=tda$comm
group=tda$group
tnst=tNST(comm=comm, group=group, rand=20,
          output.rand=TRUE, nworker=1)
# rand is usually set as 1000, here set rand=20 to save test time.

nst.pova=nst.panova(nst.result=tnst, rand=99)
# rand is usually set as 999, here set rand=99 to save test time.
```

---

null.models *Options of null model algorithms*

Description

The parameters passing to function taxo.null for each null model algorithm

Usage

```
data("null.models")
```

Format

A data frame with 13 rows on the following 3 variables. Rownames are null model algorithm IDs.

`sp.freq` character, how the species occurrence frequency will be constrainted in the null model.

`samp.rich` character, how the species richness in each sample will be constrainted in the null model.

`swap.method` character, method for fixed sp.freq and fixed samp.rich.

References

Gotelli NJ. Null model analysis of species co-occurrence patterns. Ecology 81, 2606-2621 (2000) doi:10.1890/0012-9658(2000)081[2606:nmaosc]2.0.co;2.

Examples

```
data(null.models)
```

---

pNST *Normalized Stochasticity Ratio based on phylogenetic beta diversity*

Description

Calculate normalized stochasticity ratio according to method improved from Zhou et al (2014, PNAS), based on phylogenetic beta diversity index.

Usage

```
pNST(comm, tree=NULL, pd=NULL,pd.desc=NULL,pd.wd=NULL,pd.spname=NULL,
     group, meta.group=NULL, abundance.weighted=TRUE, rand=1000,
     output.rand=FALSE, taxo.null.model=NULL, phylo.shuffle=TRUE,
     exclude.conspecifics=FALSE, nworker=4, LB=FALSE,
     between.group=FALSE, SES=FALSE, RC=FALSE)
```

Arguments

`comm`	matrix or data.frame, community data, rows are samples/sites, colnames are taxa (species/OTUs/ASVs)
`tree`	phylogenetic tree, an object of class "phylo".
`pd`	matrix, phylogenetic distance matrix.
`pd.desc`	character, the name of the file to hold the backingfile description of the phylogenetic distance matrix, it is usually "pd.desc" if using default setting in pdist.big function. If it is NULL and 'pd' is not given either, the fucntion pd.big will be used to calculate the phylogenetic distance matrix from tree, and save it in pd.wd as a big.memory file..
`pd.wd`	folder path, where the bigmemmory file of the phylogenetic distance matrix are saved.

pd.spname	character vector, taxa id in the same rank as the big matrix of phylogenetic distances.
group	a n x 1 matrix indicating the group or treatment of each sample, rownames are sample names. if input a n x m matrix, only the first column is used.
meta.group	a n x 1 matrix, to specify the metacommunity ID that each sample belongs to. NULL means the samples are from the same metacommunity.
abundance.weighted	Logic, consider abundances or not (just presence/absence). default is TRUE.
rand	integer, randomization times. default is 1000.
output.rand	Logic, whether to output dissimilarity results of each randomization. Default is FALSE.
taxo.null.model	Character, indicates null model algorithm to randomize the community matrix 'comm', including "EE", "EP", "EF", "PE", "PP", "PF", "FE", "FP", "FF", etc. The first letter indicate how to constraint species occurrence frequency, the second letter indicate how to constraint richness in each sample. see null.models for details. default is NULL, means not to randomze community but just randomize the tips, i.e. phylogeny shuffle, also named taxa shuffle.
phylo.shuffle	Logic, if TRUE, the null model algorithm "taxa shuffle" (Kembel 2009) is used, i.e. shuffling taxa labels across the tips of the phylogenetic tree to randomize phylogenetic relationships among species.
exclude.conspecifics	Logic, should conspecific taxa in different communities be exclude from MNTD calculations? default is FALSE.
nworker	for parallel computing. Either a character vector of host names on which to run the worker copies of R, or a positive integer (in which case that number of copies is run on localhost). default is 4, means 4 threads will be run.
LB	logic, whether to use a load balancing version of parallel computing code.
between.group	Logic, whether to calculate stochasticity for between-group turnovers. default is FALSE.
SES	Logic, whether to calculate standardized effect size, which is (observed dissimilarity - mean of null dissimilarity)/standard deviation of null dissimilarity. default is FALSE.
RC	Logic, whether to calculate modified Raup-Crick metric, which is percentage of null dissimilarity lower than observed dissimilarity x 2 - 1. default is FALSE.

Details

NST is a metric to estimate ecological stochasticity based on null model analysis of dissimilarity (Ning et al 2019). NST is improved from previous index ST (Zhou et al 2014). Modified stochasticity ratio (MST) is also calculated (Liang et al 2019; Guo et al 2018), which can be regarded as a spcial transformation of NST under assumption that observed similarity can be equal to mean of null similarity under pure stochastic assembly.

pNST is NST based on phylogenetic beta diversity (Ning et al 2019, Guo et al 2018), here, beta mean nearest taxon distance (bMNTD). pNST showed better performance in stochasticity estimation than tNST in some cases (Ning et al 2020).

Value

Output is a list. Please DO NOT use NST.ij values in index.pair.grp and index.between.grp which can be out of [0,1] without ecologcial meanning. Please use `nst.boot` to get variation of NST.

`index.pair` indexes for each pairwise comparison. D.ij, observed dissimilarity, not standardized; G.ij, average null expectation of dissimilarity, not standardized; Ds.ij, observed dissimilarity, standardized to range from 0 to 1; Gs.ij, average null expectation of dissimilarity, standardized; C.ij and E.ij are similarity and average null expectation of simmilarity, standardized if the dissimilarity has no fixed upper limit; ST.ij, stochasticity ratio calculated by previous method (Zhou et al 2014); MST.ij, modified stochasticity ratio calculated by a modified method (Liang et al 2019; Guo et al 2018); bNTI, beta nearest taxon index, i.e. standard effect size of difference between observed and null betaMNTD (Webb et al 2008); RC.bMNTD, modified Roup-Crick metrics (Chase et al 2011) but based on betaMNTD.

`index.grp` mean value of each index in each group. group, group name; size, number of pairwise comparisons in this group; ST.i, group mean of stochasticity ratio, not normalized; NST.i, group mean of normalized stochasticity ratio; MST.i, group mean of modified stochasticity ratio.

`index.pair.grp` pairwise values of each index in each group. group, group name; C.ij, E.ij, ST.ij, and MST.ij have the same meaning as in index.pair; NST.ij, the pairwise values of NST, for reference only, DO NOT use. Since NST is normalized ST calculated from ST.ij, NST pairwise values NST.ij have no ecological meaning. Variation of NST from bootstrapping test is preferred, see `nst.boot`.

`index.between` mean value of each index between each two groups. Similar to index.grp, but calcualted from comparisons between each two groups.

`index.pair.between` pairwise values of each index between each two groups. Similar to index.pair.grp, but calcualted from comparisons between each two groups.

`Dmax` The maximum or upper limit of dissimilarity before standardized, which is used to standardize the dissimilarity with upper limit not equal to one.

`dist.method` dissimilarity index name.

`details` detailed results. rand.mean, mean of null dissimilarity for each pairwise comparison, not standardized; Dmax, the maximum or upper limit of dissimilarity before standardized; obs3, observed dissimilarity, not standardized; dist.ran, alll null dissimilarity values, each row is a pairwise comparison, each column is results from one randomization; group, input group informaiton; meta.group, input metacommunity information.

Note

Version 3: 2020.9.9, remove setwd; change dontrun to donttest and revise save.wd in help doc. Version 2: 2020.8.22, add to NST package, update help document. Version 1: 2018.1.9

Author(s)

Daliang Ning

References

Ning D., Deng Y., Tiedje J.M. & Zhou J. (2019) A general framework for quantitatively assessing ecological stochasticity. Proceedings of the National Academy of Sciences 116, 16892-16898. doi:10.1073/pnas.1904623116.

Zhou J, Deng Y, Zhang P, Xue K, Liang Y, Van Nostrand JD, Yang Y, He Z, Wu L, Stahl DA, Hazen TC, Tiedje JM, and Arkin AP. (2014) Stochasticity, succession, and environmental perturbations in a fluidic ecosystem. Proceedings of the National Academy of Sciences of the United States of America 111, E836-E845. doi:10.1073/pnas.1324044111.

Liang Y, Ning D, Lu Z, Zhang N, Hale L, Wu L, Clark IM, McGrath SP, Storkey J, Hirsch PR, Sun B, and Zhou J. (2019) Century long fertilization reduces stochasticity controlling grassland microbial community succession. bioRxiv, doi:10.1101/638908.

Guo X, Feng J, Shi Z, Zhou X, Yuan M, Tao X, Hale L, Yuan T, Wang J, Qin Y, Zhou A, Fu Y, Wu L, He Z, Van Nostrand JD, Ning D, Liu X, Luo Y, Tiedje JM, Yang Y, and Zhou J. (2018) Climate warming leads to divergent succession of grassland microbial communities. Nature Climate Change 8, 813-818. doi:10.1038/s41558-018-0254-2.

Webb, C.O., Ackerly, D.D. & Kembel, S.W. (2008). Phylocom: software for the analysis of phylogenetic community structure and trait evolution. Bioinformatics, 24, 2098-2100.

Chase JM, Kraft NJB, Smith KG, Vellend M, and Inouye BD. (2011) Using null models to disentangle variation in community dissimilarity from variation in alpha-diversity. Ecosphere 2, art24. doi:10.1890/es10-00117.1.

Ning, D., Yuan, M., Wu, L., Zhang, Y., Guo, X., Zhou, X. et al. (2020). A quantitative framework reveals the ecological drivers of grassland soil microbial community assembly in response to warming. bioRxiv. doi:10.1101/2020.02.22.960872 (Nature Communications, under revision).

See Also

tNST, nst.boot, nst.panova

Examples

```
data("tda")
comm=tda$comm
group=tda$group
tree=tda$tree

# since it needs to save some file to a certain folder,
# the following code is set as 'not test'.
# but you may test the code on your computer
# after change the folder path for 'save.wd'.

  save.wd=tempdir() # please change to the folder you want to use.
  nworker=2 # parallel computing thread number
  rand.time=20 # usually use 1000 for real data.
  pnst=pNST(comm=comm, tree=tree, group=group,
            pd.wd=save.wd, rand=rand.time, nworker=nworker)
```

taxo.null *Null models of taxonomic beta diversity*

Description

to randomize the taxonomic structures based on one of various null model algorithms.

Usage

```
taxo.null(comm,sp.freq=c("not","equip","prop","prop.ab","fix"),
          samp.rich=c("not","equip","prop","fix"),
          swap.method=c("not","swap","tswap","quasiswap",
                        "backtrack"),burnin=0,
          abundance=c("not","shuffle","local","region"),
          region.meta=NULL)
```

Arguments

comm	matrix, community data, rownames are sample/site names, colnames are species names
sp.freq	character, the constraint of species occurrence frequency when randomizing taxonomic structures, see details.
samp.rich	character, the constraint of sample richness when randomizing taxonomic structures, see details.
swap.method	character, the swap method for fixed sp.freq and fixed samp.rich, see commsim for details.
burnin	Nonnegative integer, specifying the number of steps discarded before starting simulation. Active only for sequential null model algorithms. Ignored for non-sequential null model algorithms. also see nullmodel.
abundance	character, the method to draw individuals (abundance) into present species when randomizing taxonomic structures, see details.
region.meta	a numeric vector, to define the (relative) abundance of each species in metacommunity/regional pool. The names should be species IDs. If no name, it should be in exact the same order as columns of comm. Default is NULL, the relative abundance in metacommunity will be calculated from comm.

Details

This function returns a randomized community dataset (one time randomization), used by the function tNST. The null models differentiated by how to deal with species occurrence frequency (sp.freq), species richness in each sample (samp.rich), relative abundances (abundance), and which swap method used if both sp.freq and samp.rich are fixed.

Options of sp.freq and samp.rich (Gotelli 2000): not: the whole co-occurrence pattern (present/absent) is not randomized; equip: all the species or samples have equal probability when randomizing; prop: randomization according to probability proportional to observed species occurrence frequency or

sample richness; prop.ab: randomization according to probability proportional to observed regional abundance sum of each species, only for sp.freq; fix: randomization maintains the species occurrence frequency or sample richness exactly the same as observed.

Options of abundance: not: not abundance weighted; shuffle: randomly assign observed abundance values of observed species in a sample to species in this sample after the present/absent pattern has been randomized, thus shuffle can only be used if the richness is fixed. Similar to "richness" algorithm in R package picante (Kembel et al 2010); local: randomly draw individuals into randomized species in a sample on the probablities proportional to observed species-abundance-rank curve in this sample. If randomized species number in this sample is more than observed, the probabilities of exceeding species will be proportional to observed minimum abundance. If randomized species number (rN) in this sample is less than observed, the probabilities will be proportional to the observed abundances of top rN observed species. The rank of randomized species in a sample is randomly assigned. region: randomly draw individuals into each ranodmized species in each sample on the probabilities proportional to observed relative abundances of each species in the whole region, as described previously (Stegen et al 2013).

Value

a matrix of community data, e.g. an randomized OTU table, is returned. Rownames are sample/site names, and colnames are species names.

Note

Version 1.0: 2015.10.22

Author(s)

Daliang Ning

References

Gotelli NJ. Null model analysis of species co-occurrence patterns. Ecology 81, 2606-2621 (2000) doi:10.1890/0012-9658(2000)081[2606:nmaosc]2.0.co;2.

Kembel SW, Cowan PD, Helmus MR, Cornwell WK, Morlon H, Ackerly DD, Blomberg SP, and Webb CO. Picante: R tools for integrating phylogenies and ecology. Bioinformatics 26, 1463-1464 (2010) doi:10.1093/bioinformatics/btq166.

Stegen JC, Lin X, Fredrickson JK, Chen X, Kennedy DW, Murray CJ, Rockhold ML, and Konopka A. Quantifying community assembly processes and identifying features that impose them. Isme Journal 7, 2069-2079 (2013).

Others cited in commsim.

See Also

tNST, ab.assign, null.models

Examples

```
data(tda)
comm=tda$comm
comm.rand=taxo.null(comm,sp.freq="prop",samp.rich="fix",abundance="region")
```

---

tda *Test dataset A*

Description

A simple test data with a community matrix and treatment information

Usage

```
data("tda")
```

Format

A list object with 2 elements.

`comm` matrix, community table; each row is a sample, thus rownames are sample IDs; each column is a taxon, thus colnames are OTU IDs.

`group` matrix with only one column. treatment information; rownames are sample IDs; the only column shows treatment IDs.

`tree` phylogenetic tree.

Examples

```
data(tda)
comm=tda$comm
group=tda$group
```

---

tNST *Taxonomic Normalized Stochasticity Ratio (tNST)*

Description

Calculate normalized stochasticity ratio (NST) based on specified taxonomic dissimilarity index and null model algorithm.

Usage

```
tNST(comm, group, meta.group=NULL, meta.com=NULL,
     dist.method="jaccard", abundance.weighted=TRUE,
     rand=1000, output.rand=FALSE, nworker=4,
     LB=FALSE, null.model="PF", between.group=FALSE,
     SES=FALSE, RC=FALSE)
```

Arguments

comm	matrix or data.frame, local community data, each row is a sample or site, each colname is a species or OTU or gene, thus rownames should be sample IDs, colnames should be taxa IDs.
group	matrix or data.frame, a one-column (n x 1) matrix indicating the group or treatment of each sample, rownames are sample IDs. if input a n x m matrix, only the first column is used. Attention: different group setting will change NST values.
meta.group	matrix or data.frame, a one-column (n x 1) matrix indicating which metacommunity each sample belongs to. rownames are sample IDs. first column is metacommunity IDs. Such that different samples can belong to different metacommunities. If input a n x m matrix, only the first column is used. NULL means all samples belong to the same metacommunity. Default is NULL.
meta.com	matrix or data.frame, metacommunity data. Each row can be a samlple or a metacommunity, thus rownames are sample IDs or metacommunity IDs. Such that the relative abundance of each taxa in metacommunity can be different from the average relative abundance in the observed samples. This can be useful for uneven sampling design. NULL means the relative abundance of each taxa in the metacommunity can be directly calculated from the local community data (comm). Default is NULL.
dist.method	A character indicating dissimilarity index, including "manhattan","mManhattan", "euclidean","mEuclidean", "canberra", "bray", "kulczynski", "jaccard", "gower", "altGower", "mGower", "morisita", "horn", "binomial", "chao", "cao". default is "jaccard"
abundance.weighted	Logic, consider abundances or not (just presence/absence). default is TRUE.
rand	integer, randomization times. default is 1000
output.rand	Logic, whether to output dissimilarity results of each randomization. Default is FALSE.
nworker	for parallel computing. Either a character vector of host names on which to run the worker copies of R, or a positive integer (in which case that number of copies is run on localhost). default is 4, means 4 threads will be run.
LB	Logic, whether to use a load balancing version for parallel computation. If TRUE, this can result in better cluster utilization, but increased communication can reduce performance. default is FALSE.
null.model	Character, indicates null model algorithm, including "EE", "EP", "EF", "PE", "PP", "PF", "FE", "FP", "FF", etc. The first letter indicate how to constraint species occurrence frequency, the second letter indicate how to constraint richness in each sample. see null.models for details. default is "PF".
between.group	Logic, whether to calculate stochasticity for between-group turnovers. default is FALSE.
SES	Logic, whether to calculate standardized effect size, which is (observed dissimilarity - mean of null dissimilarity)/standard deviation of null dissimilarity. default is FALSE.
RC	Logic, whether to calculate modified Raup-Crick metric, which is percentage of null dissimilarity lower than observed dissimilarity x 2 - 1. default is FALSE.

Details

NST is a metric to estimate ecological stochasticity based on null model analysis of dissimilarity. It is improved from previous index ST (Zhou et al 2014). Detailed description is in Ning et al (2019). Modified stochasticity ratio (MST) is also calculated (Liang et al 2019; Guo et al 2018), which can be regarded as a spcial transformation of NST under assumption that observed similarity can be equal to mean of null similarity under pure stochastic assembly.

Value

Output is a list. Please DO NOT use NST.ij values in index.pair.grp and index.between.grp which can be out of [0,1] without ecologcial meanning. Please use `nst.boot` to get variation of NST.

`index.pair` indexes for each pairwise comparison. D.ij, observed dissimilarity, not standardized; G.ij, average null expectation of dissimilarity, not standardized; Ds.ij, observed dissimilarity, standardized to range from 0 to 1; Gs.ij, average null expectation of dissimilarity, standardized; C.ij and E.ij are similarity and average null expectation of simmilarity, standardized if the dissimilarity has no fixed upper limit; ST.ij, stochasticity ratio calculated by previous method (Zhou et al 2014); MST.ij, modified stochasticity ratio calculated by a modified method (Liang et al 2019; Guo et al 2018); SES.ij, standard effect size of difference between observed and null dissimilarity (Kraft et al 2011); RC.ij, modified Roup-Crick metrics (Chase et al 2011, Stegen et al 2013).

`index.grp` mean value of each index in each group. group, group name; size, number of pairwise comparisons in this group; ST.i, group mean of stochasticity ratio, not normalized; NST.i, group mean of normalized stochasticity ratio; MST.i, group mean of modified stochasticity ratio.

`index.pair.grp` pairwise values of each index in each group. group, group name; C.ij, E.ij, ST.ij, and MST.ij have the same meaning as in index.pair; NST.ij, the pairwise values of NST, for reference only, DO NOT use. Since NST is normalized ST calculated from ST.ij, NST pairwise values NST.ij have no ecological meaning. Variation of NST from bootstrapping test is preferred, see `nst.boot`.

`index.between` mean value of each index between each two groups. Similar to index.grp, but calcualted from comparisons between each two groups.

`index.pair.between` pairwise values of each index between each two groups. Similar to index.pair.grp, but calcualted from comparisons between each two groups.

`Dmax` The maximum or upper limit of dissimilarity before standardized, which is used to standardize the dissimilarity with upper limit not equal to one. See `beta.limit` for details.

`dist.method` dissimilarity index name.

`details` detailed results. rand.mean, mean of null dissimilarity for each pairwise comparison, not standardized; Dmax, the maximum or upper limit of dissimilarity before standardized; obs3, observed dissimilarity, not standardized; dist.ran, alll null dissimilarity values, each row is a pairwise comparison, each column is results from one randomization; group, input group informaiton; meta.group, input metacommunity information.

Note

Version 2: 2019.10.8. Updated references. Emphasize that NST variation should be calculated from nst.boot rather than pairwise NST.ij from tNST. Emphasize that different group setting may lead to different NST results. Version 1: 2019.5.10

Author(s)

Daliang Ning

References

Ning D., Deng Y., Tiedje J.M. & Zhou J. (2019) A general framework for quantitatively assessing ecological stochasticity. Proceedings of the National Academy of Sciences 116, 16892-16898. doi:10.1073/pnas.1904623116.

Zhou J, Deng Y, Zhang P, Xue K, Liang Y, Van Nostrand JD, Yang Y, He Z, Wu L, Stahl DA, Hazen TC, Tiedje JM, and Arkin AP. (2014) Stochasticity, succession, and environmental perturbations in a fluidic ecosystem. Proceedings of the National Academy of Sciences of the United States of America 111, E836-E845. doi:10.1073/pnas.1324044111.

Liang Y, Ning D, Lu Z, Zhang N, Hale L, Wu L, Clark IM, McGrath SP, Storkey J, Hirsch PR, Sun B, and Zhou J. (2019) Century long fertilization reduces stochasticity controlling grassland microbial community succession. bioRxiv, doi:10.1101/638908.

Guo X, Feng J, Shi Z, Zhou X, Yuan M, Tao X, Hale L, Yuan T, Wang J, Qin Y, Zhou A, Fu Y, Wu L, He Z, Van Nostrand JD, Ning D, Liu X, Luo Y, Tiedje JM, Yang Y, and Zhou J. (2018) Climate warming leads to divergent succession of grassland microbial communities. Nature Climate Change 8, 813-818. doi:10.1038/s41558-018-0254-2.

Kraft NJB, Comita LS, Chase JM, Sanders NJ, Swenson NG, Crist TO, Stegen JC, Vellend M, Boyle B, Anderson MJ, Cornell HV, Davies KF, Freestone AL, Inouye BD, Harrison SP, and Myers JA. (2011) Disentangling the drivers of beta diversity along latitudinal and elevational gradients. Science 333, 1755-1758. doi:10.1126/science.1208584.

Chase JM, Kraft NJB, Smith KG, Vellend M, and Inouye BD. (2011) Using null models to disentangle variation in community dissimilarity from variation in alpha-diversity. Ecosphere 2, art24. doi:10.1890/es10-00117.1.

Stegen JC, Lin X, Fredrickson JK, Chen X, Kennedy DW, Murray CJ, Rockhold ML, and Konopka A. (2013) Quantifying community assembly processes and identifying features that impose them. The Isme Journal 7, 2069. doi:10.1038/ismej.2013.93.

See Also

nst.boot, nst.panova, taxo.null, beta.limit, pNST

Examples

```
data(tda)
comm=tda$comm
group=tda$group
tnst=tNST(comm=comm, group=group, meta.group=NULL, meta.com=NULL,
          dist.method="jaccard", abundance.weighted=TRUE, rand=20,
```

```
          output.rand=FALSE, nworker=1, LB=FALSE, null.model="PF",
          between.group=TRUE, SES=TRUE, RC=TRUE)
# rand is usually set as 1000, here set rand=20 to save test time.
tnst.sum=tnst$NSTi
```

Index