

Package: sisters (via r-universe)

August 24, 2024

Title Runs various sister group comparisons to test hypotheses about diversification

Version 0.0.0.9000

Description Sister groups are pairs of clades that differ in a key character. If that character leads to higher diversification rate, then clades with that trait should have more species than their sisters. There are several tests to see if this is the case, ranging from a basic sign test to more complex ones (many implemented in the ape package). This package can identify sister groups that differ in a binary trait and perform all the relevant tests. It can also discretize a continuous trait into a binary ``high" vs ``low" state.

License GPL-3

Imports ape, geiger, parallel, pbapply, phangorn, stats

Suggests covr, pkgdown, testthat (>= 2.1.0)

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

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

RemoteUrl <https://github.com/bomeara/sisters>

RemoteRef HEAD

RemoteSha 219ba29741758d5a089a2f9aa6a7852d56b68250

Contents

sis_clean	2
sis_discretize	2
sis_find_taxon	3
sis_format_comparison	3
sis_format_simplified	4
sis_get_monomorphic	4
sis_get_sisters	5

sis_get_sister_pair	5
sis_get_trait_values	6
sis_iterate	6
sis_iterate_single_run	7
sis_test	8
tryNA	9

Index 10

sis_clean *Clean up trait and tree*

Description

Does basic formatting and cleanup: makes sure the taxa are the same order in both, makes sure row names of the data are taxa, etc. Relies on geiger's treedata function. The first_col_names is for software like hisse, where the first column is often taxon names.

Usage

```
sis_clean(phy, traits, first_col_names = FALSE)
```

Arguments

phy	A phylo object
traits	A data.frame of traits
first_col_names	Boolean on whether the first column has names.

Value

a list with phy and traits elements

sis_discretize *Discretize continuous trait data*

Description

Converts a vector of numbers into a vector of 0 and 1 based on whether they are below or above some value. There are two ways to do this: based on percentile or based on a numeric cutoff. By default, it will separate it based on the 50th percentile (cutoff of 0.5), but you can change the cutoff value and whether it is used as percentile or trait value.

Usage

```
sis_discretize(x, cutoff = 0.5, use_percentile = TRUE)
```

Arguments

x Vector of continuous trait values
 cutoff Value to use as cutoff. If percentile, 0.3 = 30th percentile, etc.
 use_percentile If TRUE, use cutoff as percentile

Value

a vector of 0 and 1 (and NAs)

sis_find_taxon	<i>Is the taxon in one of the sister groups</i>
----------------	---

Description

Utility function for tossing out taxa already used

Usage

```
sis_find_taxon(taxon, sisters)
```

Arguments

taxon Node number of taxon
 sisters Data.frame from sis_get_sisters()

Value

data.frame of whether the taxon is in the left or right sister group, or any

sis_format_comparison	<i>Get comparison format</i>
-----------------------	------------------------------

Description

Convert a data.frame of all sister groups (from sis_get_sisters) and a vector of 0 and 1 (with names equal to taxon names) to a data.frame with the sister groups that differ in traits.

Usage

```
sis_format_comparison(sisters, trait, phy)
```

Arguments

sisters Data.frame from sis_get_sisters()
 trait vector of 0/1 data
 phy A phylo object

Value

data.frame where each row is a sister group comparison.

Examples

```
data(geospiza, package="geiger")
cleaned <- sis_clean(geospiza$phy, geospiza$dat)
phy <- cleaned$phy
traits <- cleaned$traits
trait <- sis_discretize(traits[,1])
sisters <- sis_get_sisters(phy, ncores=2)
sisters_comparison <- sis_format_comparison(sisters, trait, phy)
print(sisters_comparison)
```

sis_format_simplified *Get simplified comparison format suitable for passing into other functions*

Description

Get simplified comparison format suitable for passing into other functions

Usage

```
sis_format_simplified(sisters_comparison)
```

Arguments

sisters_comparison
Data.frame from sis_format_comparison

Value

Data.frame of two columns: diversity with state 0 and state 1, where each row is a sister group comparison

sis_get_monomorphic *Get monomorphic trait*

Description

Get monomorphic trait

Usage

```
sis_get_monomorphic(trait)
```

Arguments

trait Vector of trait values

Value

The state all taxa have if monomorphic; NA otherwise

sis_get_sisters *Get sister groups for all internal nodes*

Description

For each node, return the vector of tip numbers for taxa on each side. It is sorted so that sister groups with fewer taxa are arranged at the top.

Usage

```
sis_get_sisters(phy, ncores = 2)
```

Arguments

phy A phylo object
ncores How many cores to use to run this in parallel. I suggest `parallel::detectCores()`, but set it at 2 for a default (otherwise CRAN checks fail)

Value

a data.frame with the node numbers and columns with the tip labels of the two descendant clades, plus additional info on the sister groups

sis_get_sister_pair *Get sister groups for a node*

Description

For a node, gives the taxa on each side. Note that the output is a data.frame with lists

Usage

```
sis_get_sister_pair(node, phy)
```

Arguments

node Node number
phy A phylo object

Value

a data.frame with the node numbers and columns with the tip labels of the two descendant clades

sis_get_trait_values *Get trait values for tip numbers*

Description

Get trait values for tip numbers

Usage

```
sis_get_trait_values(nodes, phy, trait)
```

Arguments

nodes	vector of node numbers (tip numbers, actually)
phy	A phylo object
trait	A trait vector with names equal to taxon names

sis_iterate *Iterate tests trying a variety of cutoff values*

Description

This is a way of looking at the effect of using different cutoff values on the sister group comparisons. Do clades with a higher value have more species than their sister, and is this robust to what cutoff value is used? At the extremes (the min and max value) this is almost certainly not the case, unless you have many taxa with the same maximum or minimum values.

Usage

```
sis_iterate(
  x,
  nsteps = 11,
  phy,
  sisters = sis_get_sisters(phy),
  drop_matches = TRUE
)
```

Arguments

x	Vector of continuous trait values
nsteps	Number of thresholds to try
phy	A phylo object
sisters	Data.frame from sis_get_sisters()
drop_matches	Drop sister group comparisons with equal numbers of taxa

Details

This is a very dangerous function to use. Someone could use this to find the perfect cutoff value to find a significant result. This is one of the many forms of p-hacking. **So, if you use this function and then report on significance using some cutoff, you MUST mention somewhere in your manuscript that you've tried a variety of cutoff values, and include a discussion of why you used a particular cutoff.** Ideally, you should have some biological intuition about what cutoff value is reasonable before using this function, as well.

Value

A data.frame, where each column is for a different cutoff percentile and every row is a number returned from sis_test()

Examples

```
data(geospiza, package="geiger")
cleaned <- sis_clean(geospiza$phy, geospiza$dat)
phy <- cleaned$phy
trait <- cleaned$traits[,1]
sis_iterate(trait, phy=phy)
```

sis_iterate_single_run

Do a test with a single cutoff value

Description

Do a test with a single cutoff value

Usage

```
sis_iterate_single_run(
  cutoff,
  x,
  use_percentile = TRUE,
  phy,
  sisters = sis_get_sisters(phy),
  drop_matches = TRUE,
  warn = FALSE
)
```

Arguments

cutoff	Value to use as cutoff. If percentile, 0.3 = 30th percentile, etc.
x	Vector of continuous trait values
use_percentile	If TRUE, use cutoff as percentile
phy	A phylo object
sisters	Data.frame from sis_get_sisters()
drop_matches	Drop sister group comparisons with equal numbers of taxa
warn	Some tests will fail with warnings (too few sister groups or other reasons). Setting this to FALSE will suppress those

Value

vector of output from sis_test()

sis_test	<i>Compute multiple tests based on sister group comparisons</i>
----------	---

Description

Compute multiple tests based on sister group comparisons

Usage

```
sis_test(pairs, drop_matches = TRUE, warn = TRUE)
```

Arguments

pairs	Data.frame with one row per sister group comparison, with one column for number of taxa in state 0, and one column for the number of taxa in state 1.
drop_matches	Drop sister group comparisons with equal numbers of taxa
warn	Some tests will fail with warnings (too few sister groups or other reasons). Setting this to FALSE will suppress those

Value

A vector with the results of many tests, as well as summary data for the comparisons

Examples

```
data(geospiza, package="geiger")
cleaned <- sis_clean(geospiza$phy, geospiza$dat)
phy <- cleaned$phy
traits <- cleaned$traits
trait <- sis_discretize(traits[,1])
sisters <- sis_get_sisters(phy)
sisters_comparison <- sis_format_comparison(sisters, trait, phy)
pairs <- sis_format_simplified(sisters_comparison)
sis_test(pairs)
```

tryNA	<i>Try but returns NA rather than error</i>
-------	---

Description

Try but returns NA rather than error

Usage

```
tryNA(code, silent = FALSE)
```

Arguments

code	Code to run
silent	Print error if TRUE

Index

sis_clean, 2
sis_discretize, 2
sis_find_taxon, 3
sis_format_comparison, 3
sis_format_simplified, 4
sis_get_monomorphic, 4
sis_get_sister_pair, 5
sis_get_sisters, 5
sis_get_trait_values, 6
sis_iterate, 6
sis_iterate_single_run, 7
sis_test, 8

tryNA, 9