Package 'philr'

November 12, 2026

Type Package

Title Phylogenetic partitioning based ILR transform for metagenomics data

Version 1.38.0

Date 2026-04-01

Description PhILR is short for Phylogenetic Isometric Log-Ratio Transform. This package provides functions for the analysis of compositional data (e.g., data representing proportions of different variables/parts). Specifically this package allows analysis of compositional data where the parts can be related through a phylogenetic tree (as is common in microbiota survey data) and makes available the Isometric Log Ratio transform built from the phylogenetic tree and utilizing a weighted reference measure.

License GPL-3

RoxygenNote 7.3.3

Imports ape, phangorn, tidyr, ggplot2, ggtree, methods

Suggests testthat, knitr, ecodist, rmarkdown, BiocStyle, phyloseq, SummarizedExperiment, TreeSummarizedExperiment, glmnet, dplyr, mia

VignetteBuilder knitr

biocViews ImmunoOncology, Sequencing, Microbiome, Metagenomics, Software

URL https://github.com/jsilve24/philr

BugReports https://github.com/jsilve24/philr/issues

Encoding UTF-8

git_url https://git.bioconductor.org/packages/philr

git_branch RELEASE_3_23

git_last_commit 5603163

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Author Justin Silverman [aut, cre],
Leo Lahti [ctb] (ORCID: <https://orcid.org/0000-0001-5537-637X>)

Maintainer Justin Silverman <jsilve24@gmail.com>

Contents

annotate_balance 2
buildilrBasep 4
calculate.blw 4
clrp 5
convert_to_long 7
g.colMeans 8
g.rowMeans 8
ilrp 9
mean_dist_to_tips 10
miniclo 11
name.balance 12
named_rtree 13
name_nodenumber_conversion 14
philr 15
philrInv 18
phylo2sbp 19
shift 20

Index **22**

`annotate_balance` *annotate_balance*

Description

annotate a balance oriented with respect to the PhILR transform. That is, you can specify labels for the numerator (up) and denominator (down).

Usage

```
annotate_balance(
  tr,
  coord,
  p = NULL,
  labels = c("+", "-"),
  offset = 0,
  offset.text = 0.03,
  bar = TRUE,
  barsize = 0.01,
  barfill = "darkgrey",
  geom = "text",
```

```
  ...
)
```

Arguments

tr	phylo object
coord	named internal node/balance to annotate
p	ggtree plot (tree layer), if NULL then a new plot will be created.
labels	label for the numerator and denominator of the balance respectively
offset	offset for bar (if bar=TRUE) from tips
offset.text	offset of text from bar (if bar=TRUE) or from tips (if bar=FALSE)
bar	logical, should bar for each clade be plotted
barsize	width of bar (if bar=TRUE)
barfill	fill of bar
geom	geom used to draw label (e.g., 'text' or 'label')
...	additional parameters passed to geom_rect and specified geom

Value

ggplot object

Author(s)

Justin Silverman

References

Guangchuang Yu, David Smith, Huachen Zhu, Yi Guan, Tommy Tsan-Yuk Lam. *ggtree: an R package for visualization and annotation of phylogenetic trees with their covariates and other associated data.* Methods in Ecology and Evolution 2016, doi:10.1111/2041-210X.12628

Examples

```
tr <- named_rtree(10)

annotate_balance(tr, 'n4', size=7)
annotate_balance(tr, 'n4', size=7, barsize=0.04, barfill='darkgreen',
    offset.text=0.05, color='red')
annotate_balance(tr, 'n4', bar=FALSE, size=7)
annotate_balance(tr, 'n4', bar=TRUE, size=7, labels=c('Num', 'Denom'),
    offset.text=.3)
annotate_balance(tr, 'n4', bar=TRUE, geom='label', size=8, offset.text=0.1)
```

buildilrBasep *Weighted ILR Contrast Matrix*

Description

Weighted ILR Contrast Matrix

Usage

```
buildilrBasep(W, p)
```

Arguments

W	sequantial binary partition matrix (e.g., signary matrix; output of phylo2sbp)
p	weights (should not be closed)

Value

matrix

Author(s)

Justin Silverman (adapted from compositions::gsi.buildilrBase)

References

J. J. Egozcue, V. Pawlowsky-Glahn (2016) *Changing the Reference Measure in the Simplex and its Weighting Effects*. Austrian Journal of Statistics 45(4):25-44

Examples

```
p <- seq(.1,1,by=.2)
tr <- named_rtree(5)
sbp <- phylo2sbp(tr)
buildilrBasep(sbp, p)
```

calculate.blw *Calculate Branch Length Weightings for ILR Coordinates*

Description

Calculates the weightings for ILR coordinates based on branch lenghts of a phylogenetic tree via a few different methods (see details).

Usage

```
calculate.blw(tree, method = "sum.children")
```

Arguments

tree	a phylo class tree object that is binary (see multi2di)
method	options include: (default) 'sum.children' and 'mean.descendants' see details for more information.

Details

ILR balances built from a binary partition of a phylogenetic tree can be imbued with branch length information. This function is helpful in calculating those weightings.

There are a number of methods for calculating these weightings, the default 'sum.children' calculates the weighting for a given balance as the sum of its two direct children's branch length. An alternative that has been as yet less studied is 'mean.descendants' to calculate the weighting for a given balance as the sum of its two direct children's branch lengths PLUS for each child the average distance from it to its descendant tips.

Note: That some trees contain tips with branch lengths of zero length. This can result in that tip being unreasonably downweighted, as such this function automatically adds a small pseudocount to those tips with zero length (equal to the smallest non-zero) branch length on the tree.

Value

vector of weightings for ILR coordinates produced via specified method.

Author(s)

Justin Silverman

See Also

philr

Examples

```
tr <- named_rtree(50)
calculate.blw(tr, method='sum.children')[1:10]
calculate.blw(tr, method='mean.descendants')[1:10]
```

---

clrp *Weighted CLR Transform*

Description

Weighted CLR Transform

Usage

```
clrp(y, p)

clrpInv(y.star)
```

Arguments

y	shifted data matrix (e.g., output of shiftp)
p	weights (should not be closed)
y.star	a data matrix that represents data transformed by clrp

Details

Note that this function will close the dataset y to 1.

Inverting clrp transform should be followed by shiftpInv to return to unshifted original compositoin (see examples).

Value

matrix

Author(s)

Justin Silverman

References

J. J. Egozcue, V. Pawlowsky-Glahn (2016) *Changing the Reference Measure in the Simplex and its Weighting Effects*. Austrian Journal of Statistics 45(4):25-44

Examples

```
p <- seq(.1,1,by=.2)
c <- t(rmultinom(10,100,c(.1,.6,.2,.3,.2))) + 0.65   # add a small pseudocount
x <- miniclo(c)
y <- shiftp(x, p)
y.star <- clrp(y, p)
y.star

# Untransform data (note use of shiftp and miniclo to return to x)
y.closed <- clrpInv(y.star)
all.equal(miniclo(shiftpInv(y.closed, p)), x)
```

convert_to_long *Converts wide format ILR transformed data to long format*

Description

Converts wide format ILR transformed data (see philr) to long format useful in various plotting functions where long format data is required.

Usage

```
convert_to_long(x, labels)
```

Arguments

x	PhILR transformed data in wide format (samples by balances) (see philr)
labels	vector (of length nrow(x)) with labels to group samples by

Value

x in long format with columns

- sample
- labels
- coord
- value

Examples

```
tr <- named_rtree(5)
x <- t(rmultinom(10,100,c(.1,.6,.2,.3,.2))) + 0.65 # add small pseudocount
colnames(x) <- tr$tip.label

x.philr <- philr(x, tree=tr, part.weights='enorm.x.gm.counts',
   ilr.weights='blw.sqrt', return.all=FALSE)
   convert_to_long(x.philr, rep(c('a','b'), 5))
```

g.colMeans *Geometric Means of Columns*

Description

Calculates geometric mean of columns. Does not calculate WEIGHTED geometric means (vs. g.rowMeans)

Usage

```
g.colMeans(x)
```

Arguments

x	matrix or vector

Value

vector (geometric mean of columns)

See Also

g.rowMeans

Examples

```
philr:::g.colMeans(rbind(c(2,4,4), c(2,4,4)))
```

g.rowMeans *Weighted Geometric Means of Rows*

Description

Calculates weighted geometric mean (see references). Note if `p=rep(1, nrow(y))` (default) then this is just the geometric mean of rows.

Usage

```
g.rowMeans(y, p = rep(1, nrow(y)))
```

Arguments

y	shifted data matrix (e.g., output of shiftp)
p	weights (should not be closed)

Value

vector (weighted geometric mean of rows)

References

J. J. Egozcue, V. Pawlowsky-Glahn (2016) *Changing the Reference Measure in the Simplex and its Weighting Effects*. Austrian Journal of Statistics 45(4):25-44

See Also

g.colMeans

Examples

```
p <- seq(.1,1,by=.2)
c <- t(rmultinom(10,100,c(.1,.6,.2,.3,.2))) + 0.65   # add a small pseudocount
x <- miniclo(c)
y <- shiftp(x, p)
philr:::g.rowMeans(y, p)
```

---

ilrp *Weighted ILR Transform*

Description

Calculated using weighted CLR transform (clrp)

Usage

```
ilrp(y, p, V)

ilrpInv(y.star, V)
```

Arguments

y	shifted data matrix (e.g., output of shiftp)
p	weights (should not be closed)
V	weighted contrast matrix (e.g., output of buildilrBasep)
y.star	a data matrix that represents data transformed by ilrp

Value

matrix

Author(s)

Justin Silverman

References

J. J. Egozcue, V. Pawlowsky-Glahn (2016) *Changing the Reference Measure in the Simplex and its Weighting Effects*. Austrian Journal of Statistics 45(4):25-44

See Also

philrInv

Examples

```
# Weights
p <- seq(.1,1,by=.2)

# Shifted Composition
c <- t(rmultinom(10,100,c(.1,.6,.2,.3,.2))) + 0.65   # add a small pseudocount
x <- miniclo(c)
y <- shiftp(x, p)

# Contrast Matrix
tr <- named_rtree(5)
sbp <- phylo2sbp(tr)
V <- buildilrBasep(sbp, p)

y.star <- ilrp(y, p, V)
y.star

# Untransform data (note use of shiftp and miniclo to return to x)
y.closed <- ilrpInv(y.star, V)
all.equal(miniclo(shiftpInv(y.closed, p)), x, check.attributes=FALSE)
```

---

mean_dist_to_tips *Mean distance from internal nodes to descendant tips*

Description

Calculates the mean distance from each internal node to its descendant tips

Usage

```
mean_dist_to_tips(tree)
```

Arguments

tree	a phylo class tree object that is binary (see multi2di)

Details

This is a function used by calculate.blw when method='mean.descendants', there this function is called twice, once for each direct child of a given internal node and the results are summed for each node.

Value

vector (named if internal nodes are named)

Examples

```
tr <- named_rtree(5)
mean_dist_to_tips(tr)
```

`miniclo` *miniclo*

Description

small function to close (aka normalize by proportions, aka total sum scaling) a dataset to a constant k (usually taken to be 1). After closure the row sums of the dataset should sum to k.

Usage

```
miniclo(c, k = 1)
```

Arguments

c	dataset to be closed
k	closure constant

Value

matrix (if c is a vector or matrix) or data.frame (if c is a data.frame)

Examples

```
c <- matrix(c(1,2,3,1,2,3,1,2,3), nrow = 3, byrow=TRUE)
miniclo(c)
miniclo(c, k=2)
```

`name.balance` *Name a balance (coordinate) based on taxonomy*

Description

For a given ILR balance (coordinate) assigns a name to the balance based on a provided taxonomy table. This is useful for interpretation of the balances.

Usage

```
name.balance(
  tr,
  tax,
  coord,
  method = "voting",
  thresh = 0.95,
  return.votes = NULL
)
```

Arguments

`tr`	an object of class `'phylo'`
`tax`	a matrix/data.frame of taxonomy, rownames should correspond to `tr$tip.labels` columns should be taxonomic levels (named) with increasing taxonomic resolution from left to right (e.g., Phylum to the left of Genus).
`coord`	the name of a balance/internal node on the tree (given as a string)
`method`	currently only `'voting'` implemented. See Details.
`thresh`	threshold for assignment of taxonomy to a given part of a balance (must be greater than 0.5 if `method='voting'`; see details).
`return.votes`	whether voting results by taxonomic level should be shown for `coord`. Note: this is helpful when `name.balance` does not return a clear winner, as may be the case when a given `coord` represents more than one taxonomic lineage. votes are returned as a list indexed by `colnames(tax)` Options include: `NULL` (default) only returns the combined consensus name of the balance `'up'` adds tallied votes for the 'up' node to the output list `'down'` adds tallied votes for the 'down' node to the output list `'self'` adds tallied votes for `coord` to the output list

Details

A bit of terminology:

coord this is the same as the names of the balances which should be the same as the names of the internal nodes of `tr`

'up' this is the child node of `coord` that is represented in the numerator of the `coord` balance.

'down' this is the child node of `coord` that is represented in the denominator of the `coord` balance

The method `'voting'` assigns the name of the each part of a balance (e.g., numerator and denominator / each child of `coord`) as follows:

1. First Subset `tax` to contain only descendent tips of the given child of `coord`
2. Second At the finest taxonomic (farthest right of `tax`) see if any one taxonomic label is present at or above `thresh`. If yes output that taxonomic label (at that taxonomic level) as the label for that child of `coord`. If no then move to coarser taxonomic level (leftward) and repeat.

Value

If `return.votes=NULL` returns a string of the form (ex. 'Genus_Bacteroides/Phylum_Firmicutes'). Otherwise returns a list with the above string as 'name', see Arguments for `show.votes` for other optional returned items.

Author(s)

Justin Silverman

See Also

`philr`

Examples

```
tr <- named_rtree(40)
tax <- data.frame(Kingdom=rep('A', 40),
                  Phylum=rep(c('B','C'), each=20),
                  Genus=c(sample(c('D','F'),20, replace=TRUE),
                          sample(c('G','H'), 20, replace=TRUE)))
rownames(tax) <- tr$tip.label
name.balance(tr, tax, 'n1')
name.balance(tr, tax, 'n34')
name.balance(tr,tax, 'n34', return.votes = c('up', 'down'))
```

---

named_rtree *Generate random tree with named internal nodes*

Description

Internal nodes are named by numbering and adding the prefix 'n'. This function is laregly for use in examples throughout this package.

Usage

```
named_rtree(n)
```

Arguments

n	an integer giving the number of tips in the tree.

Value

An object of class "phylo"

Examples

```
named_rtree(5)
```

---

name_nodenumber_conversion

Convert between node/tip labels and integer node numbers

Description

Useful if you want to convert between node labels (c), tip labels (t) and the internal integer number that identifies that node (nn). Particularly for use with plotting libraries.

Usage

```
nn.to.name(tr, x)

name.to.nn(tr, x)
```

Arguments

tr	object of type phylo
x	vector of numerics or characters

Value

vector

Examples

```
tr <- named_rtree(5)
name.to.nn(tr, 'n1')
name.to.nn(tr,c('n1','n2','t1'))
nn.to.name(tr, 1:9)
```

philr *Data transformation and driver of PhILR.*

Description

This is the main function for building the phylogenetic ILR basis, calculating the weightings (of the parts and the ILR coordinates) and then transforming the data.

Usage

```
philr(
  x,
  tree = NULL,
  sbp = NULL,
  part.weights = "uniform",
  ilr.weights = "uniform",
  return.all = FALSE,
  pseudocount = 0,
  abund_values = "counts",
  ...
)
```

Arguments

x	**matrix** of data to be transformed (samples are rows, compositional parts are columns). Should be provided as **raw counts** – do not pre-normalize to relative abundances (see Details). Zeros must be handled prior to calling philr via pseudocount, multiplicative replacement, or another method (see also the pseudocount argument).
tree	a phylo class tree object that is binary (see multi2di)
sbp	(Optional) give a precomputed sbp matrix phylo2sbp if you are going to build multiple ILR bases (e.g., with different weightings).
part.weights	weightings for parts, can be a named vector with names corresponding to colnames(x) otherwise can be a string, options include: 'uniform' (default) uses the uniform reference measure 'gm.counts' geometric mean of parts of x 'anorm' aitchison norm of parts of x (after closure) 'anorm.x.gm.counts' 'anorm' times 'gm.counts' 'enorm' euclidean norm of parts of x (after closure) 'enorm.x.gm.counts' 'enorm' times 'gm.counts', often gives good results
ilr.weights	weightings for the ILR coordiantes can be a named vector with names corresponding to names of internal nodes of tree otherwise can be a string, options include: 'uniform' (default) no weighting of the ILR basis

'blw' sum of children's branch lengths

'blw.sqrt' square root of 'blw' option

'mean.descendants' sum of children's branch lengths PLUS the sum of each child's mean distance to its descendent tips

return.all return all computed parts (e.g., computed sign matrix(sbp), part weightings (p), ilr weightings (ilr.weights), contrast matrix (V)) as a list (default=FALSE) in addition to in addition to returning the transformed data (.ilrp). If return.all==FALSE then only returns the transformed data (not in list format) If FALSE then just returns list containing x.ilrp.

pseudocount optional pseudocount added to observation matrix (‘x‘) to avoid numerical issues from zero values. Default value is 0 which has no effect (allowing the user to handle zeros in their own preffered way before calling philr). Values < 0 given an error.

abund_values A single character value for selecting the assay to be used. Only used when x is object from this class. Default: "counts".

... other parameters passed to philr.data.frame or philr.TreeSummarizedExperiment

Details

This is a utility function that pulls together a number of other functions in philr. The steps that are executed are as follows:

1. Create sbp (sign matrix) if not given
2. Create parts weightings if not given
3. Shift the dataset with respect to the new reference measure (e.g., part weightings)
4. Create the basis contrast matrix from the sign matrix and the reference measure
5. Transform the data based on the contrast matrix and the reference measure
6. Calculate the specified ILR weightings and multiply each balance by the corresponding weighting

Note for both the reference measure (part weightings) and the ILR weightings, specifying 'uniform' will give the same results as not weighting at all.

Input format: x should be provided as raw counts (after zero handling via pseudocount or multiplicative replacement) and *not* pre-normalized to relative abundances. Closure is applied internally. This matters because several part.weights options ('gm.counts', 'anorm.x.gm.counts', 'enorm.x.gm.counts') are computed from x prior to closure and use count magnitudes to weight parts; pre-normalizing to relative abundances removes this information and alters the intended weighting. When part.weights='uniform' (the default), the ILR is scale-invariant and counts or relative abundances yield identical results, but passing counts is still recommended for consistency.

The tree argument is ignored if the x argument is assay or assay These objects can include a phylogenetic tree. If the phylogenetic tree is missing from these objects, it should be integrated directly in these data objects before running philr. Alternatively, you can always provide the abundance matrix and tree separately in their standard formats.

If you have a assay, this can be converted into assay, to incorporate tree information.

Value

matrix if `return.all=FALSE`, if `return.all=TRUE` then a list is returned (see above).

Author(s)

Justin Silverman; S3 methods by Leo Lahti

See Also

`phylo2sbp` `calculate.blw`

Examples

```
# Prepare example data
tr <- named_rtree(5)
x <- t(rmultinom(10,100,c(.1,.6,.2,.3,.2))) + 0.65 # add a small pseudocount
colnames(x) <- tr$tip.label
philr(x, tr, part.weights='enorm.x.gm.counts',
               ilr.weights='blw.sqrt', return.all=FALSE)

# Running philr on a TreeSummarizedExperiment object

## Prepare example data
library(mia)
library(tidyr)
data(GlobalPatterns, package="mia")

## Select prevalent taxa
tse <-  GlobalPatterns %>% subsetByPrevalentTaxa(
                               detection = 3,
                               prevalence = 20/100,
                               as_relative = FALSE)

## Pick taxa that have notable abundance variation across sammples
variable.taxa <- apply(assay(tse, "counts"), 1, function(x) sd(x)/mean(x) > 3.0)
tse <- tse[variable.taxa,]

# Collapse the tree
tree <- ape::keep.tip(phy = rowTree(tse), tip = rowLinks(tse)$nodeNum)
rowTree(tse) <- tree

## Add a new assay with a pseudocount
assays(tse)$counts.shifted <- assay(tse, "counts") + 1

## Run philr for TreeSummarizedExperiment object
## using the pseudocount data
res.tse <- philr(tse, part.weights='enorm.x.gm.counts',
               ilr.weights='blw.sqrt', return.all=FALSE,
               abund_values="counts.shifted")

# Running philr on a phyloseq object
## Not run:
```

```
  pseq <- convertToPhyloseq(tse)
  res.pseq <- philr(pseq, part.weights='enorm.x.gm.counts',
              ilr.weights='blw.sqrt', return.all=FALSE,
              pseudocount=0.5)

## End(Not run)
```

---

philrInv *Inverse of PhILR Transform*

Description

Inverse of PhILR Transform

Usage

```
philrInv(
  x.ilrp,
  tree = NULL,
  sbp = NULL,
  V = NULL,
  part.weights = NULL,
  ilr.weights = NULL
)
```

Arguments

x.ilrp	transformed data to which the inverse transform will be applied
tree	(optional) to be used to build sbp and contrast matrix (see details)
sbp	(optional) the sbp (sequential binary partition) used to build a contrast matrix (see details)
V	(optional) the contrast matrix (see details)
part.weights	weightings for parts, can be a named vector with names corresponding to colnames(x). Defaults to 'uniform' (part.weights = 1,...,1)
ilr.weights	weightings for the ILR coordiantes can be a named vector with names corresponding to names of internal nodes of tree. Defaults to 'uniform' (ilr.weights = 1,...,1)

Details

This is a utility function for calculating the inverse of the philr transform. Note that at least one of the following parameters must be specified (tree, sbp, or V).

Value

a matrix of compositions (rows are samples, columns are parts), function removes the effects of ilr weights, part weights, and unshifts the composition.

Author(s)

Justin Silverman

See Also

philr

Examples

```
tr <- named_rtree(5)
x <- t(rmultinom(10,100,c(.1,.6,.2,.3,.2))) + 0.65 # add small pseudocount
colnames(x) <- tr$tip.label
d <- philr(x, tr, part.weights='enorm.x.gm.counts',
               ilr.weights='blw.sqrt', return.all=TRUE)
d.inverted <- philrInv(d$x.ilrp, V=d$V, part.weights = d$p,
                       ilr.weights = d$ilr.weights)
all.equal(miniclo(x), d.inverted)
```

---

phylo2sbp *Create Sequential Binary Partition from Phylogenetic Tree*

Description

This function converts a binary phylogenetic tree to sequential binary parition to be used to then build an ILR basis for compositional metagenomic data.

Usage

```
phylo2sbp(tr)
```

Arguments

tr	a phylo tree object with n leaves

Details

The choice of orientation for a balance (i.e., which of the two descendant clades of an internal node is in the numerator or denominator of the log-ratio) is given by the default of the function phangorn::Children and that choise is used consistently throughout the philr package.

Value

a n by n-1 matrix of the sequential binary partition sign matrix

Author(s)

Justin Silverman

References

Schliep K.P. 2011. phangorn: phylogenetic analysis in R. Bioinformatics, 27(4) 592-593

See Also

philr

Examples

```
tr <- named_rtree(5)
phylo2sbp(tr)
```

`shift` *Shift data to origin given by p*

Description

Shift must be applied before transformation

Usage

```
shiftp(x, p)

shiftpInv(y, p)
```

Arguments

`x`	closed compositional data matrix (or vector)
`p`	weights (should not be closed)
`y`	the shifted composition output by `shiftp`

Value

shifted data matrix `y` (no closure is applied) rows are samples, columns are parts

Author(s)

Justin Silverman & J. J. Egozcue

References

J. J. Egozcue, V. Pawlowsky-Glahn (2016) *Changing the Reference Measure in the Simplex and its Weighting Effects*. Austrian Journal of Statistics 45(4):25-44

Examples

```
p <- seq(.1,1,by=.2)
c <- t(rmultinom(10,100,c(.1,.6,.2,.3,.2))) + 0.65   # add a small pseudocount
x <- miniclo(c)
shiftp(x, p)
```

Index

annotate_balance, 2
assay, *16*

build.phylo.ilr *(*philr*)*, 15
buildilrBasep, 4, *9*

calc.blw *(*calculate.blw*)*, 4
calculate.blw, 4, *10*, *17*
clrp, 5, *9*
clrpInv *(*clrp*)*, 5
convert_to_long, 7

g.colMeans, 8, *9*
g.rowMeans, *8*, 8

ilrp, 9
ilrpInv *(*ilrp*)*, 9

mean_dist_to_tips, 10
miniclo, 11
multi2di, *5*, *10*, *15*

name.balance, 12
name.to.nn
        *(*name_nodenumber_conversion*)*,
        14
name_nodenumber_conversion, 14
named_rtree, 13
nn.to.name
        *(*name_nodenumber_conversion*)*,
        14

philr, *5*, *7*, *13*, 15, *18–20*
philrInv, *10*, 18
phylo2sbp, *4*, *15*, *17*, 19

shift, 20
shiftp, *6*, *8*, *9*
shiftp *(*shift*)*, 20
shiftpInv *(*shift*)*, 20