Package 'hilldiv3'

November 6, 2026

Title Integral Analysis of Diversity Based on Hill Numbers

Version 3.2.1

Description Measures and compares the diversity of biological communities (e.g. tables of operational taxonomic units (OTUs), amplicon sequence variants (ASVs) or metagenome-assembled genomes (MAGs)) based on Hill numbers, in a unified framework for neutral, phylogenetic and functional diversity measurement, diversity partitioning, (dis)similarity measurement, diversity profiles, evenness, redundancy and attribution of Hill power-sum weight and beta dissimilarity to taxa and sets. The statistical framework encompasses richness, Shannon and Simpson diversity, Faith's phylogenetic diversity (PD), Rao's quadratic entropy and Sorensen- and UniFrac-type dissimilarities, all grounded in a single Hill-number framework. Methods are described in Jost (2007) <doi:10.1890/06-1736.1>, Chao et al. (2010) <doi:10.1098/rstb.2010.0272>, Chiu et al. (2014) <doi:10.1890/12-0960.1> and reviewed in Alberdi & Gilbert (2019) <doi:10.1111/1755-0998.13014>. Optional import adapters interoperate with the Bioconductor packages 'phyloseq', 'SummarizedExperiment' and 'TreeSummarizedExperiment', which are available from <https://bioconductor.org>.

License GPL-3

Language en-GB

URL https://github.com/alberdilab/hilldiv3, https://alberdilab.github.io/hilldiv3/

BugReports https://github.com/alberdilab/hilldiv3/issues

Depends R (>= 4.1.0)

Imports ape, cli, grDevices, graphics, methods, rlang, stats, utils

Suggests cluster, furrr, future, ggplot2, knitr, patchwork, phyloseq, progressr, rmarkdown, SummarizedExperiment, testthat (>= 3.0.0), tibble, TreeSummarizedExperiment, vegan

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

LazyData true

RoxygenNote 7.3.3

NeedsCompilation no

Author Antton Alberdi [aut, cre] (ORCID:
<https://orcid.org/0000-0002-2875-6446>)

Maintainer Antton Alberdi <antton.alberdi@sund.ku.dk>

Repository CRAN

Date/Publication 2026-11-06 05:00:11 UTC

Contents

gut_counts . . . 2
gut_traits . . . 3
gut_tree . . . 4
hillcontrib . . . 4
hilldiss . . . 6
hilldiv . . . 8
hilleven . . . 10
hillpair . . . 11
hillpart . . . 13
hillprof . . . 15
hillred . . . 16
hillshare . . . 18
hillsim . . . 19
match_data . . . 21
plot.hill_profile . . . 22
plot.hill_redundancy . . . 22
traits2dist . . . 23
tss . . . 24

Index **25**

`gut_counts` *Simulated gut microbiome MAG count table*

Description

A small, **simulated** example data set representing metagenome-assembled genome (MAG) abundances across host samples from two groups (`control` and `treatment`), where a block of MAGs is enriched in the treatment group so that beta diversity is non-trivial. Generated by `data-raw/make-data.R`.

Usage

```
gut_counts
```

Format

An integer matrix with 24 rows (MAGs, `mag01`..`mag24`) and 12 columns (samples, `ctrl01`..`ctrl06` and `trt01`..`trt06`).

See Also

gut_tree, gut_traits

Examples

```
hilldiv(gut_counts, q = c(0, 1, 2))
```

---

gut_traits *Functional traits for the simulated gut MAGs*

Description

A trait table for the 24 MAGs in gut_counts, mixing continuous, categorical and binary traits. Convert it to a functional distance with `traits2dist()` for the functional-diversity paths.

Usage

```
gut_traits
```

Format

A data frame with 24 rows (MAGs) and 4 columns:

genome_size Approximate genome size in Mbp (numeric).

gc_content GC content as a proportion (numeric).

oxygen Oxygen tolerance: aerobe, anaerobe or facultative (factor).

motility Motility indicator, 0/1 (integer).

See Also

gut_counts, gut_tree, `traits2dist()`

Examples

```
d <- traits2dist(gut_traits)
hilldiv(gut_counts, q = c(0, 1), dist = d)
```

gut_tree *Phylogeny for the simulated gut MAGs*

Description

An ultrametric coalescent tree over the 24 MAGs in gut_counts, scaled to unit depth. Use it for the phylogenetic-diversity paths.

Usage

```
gut_tree
```

Format

A `phylo` object (see the **ape** package) with 24 tips whose labels match the rows of gut_counts.

See Also

gut_counts, gut_traits

Examples

```
hilldiv(gut_counts, q = c(0, 1), tree = gut_tree)
```

hillcontrib *Attribute Hill dissimilarity to taxa or sets*

Description

Show which taxa contribute most to dissimilarity between every pair of samples or among samples in a group. Default singleton-taxon contributions are nonnegative and add up to the corresponding value from `hillpair()` or `hilldiss()`. A taxon's `share` is its fraction of that dissimilarity; it is `NA` when the dissimilarity is zero.

Usage

```
hillcontrib(
  data,
  q = c(0, 1, 2),
  metric = c("S", "C", "U", "V"),
  tree = NULL,
  dist = NULL,
  tau = NULL,
  type = c("auto", "neutral", "phylogenetic", "functional"),
  by = c("pairwise", "collective"),
  groups = NULL,
  sets = NULL
)
```

Arguments

data	A count table (taxa x samples) or a supported object; a single sample is not meaningful for partitioning.
q	Numeric vector of diversity orders (>= 0). Defaults to c(0, 1, 2) (richness, Shannon, Simpson).
metric	Dissimilarity metric(s) to return, any of "S", "C", "U", "V". Defaults to all four. "S" (Jaccard-type) and "V" (Sorensen-type) are turnover measures, relative to the pooled assemblage and to a single assemblage respectively; "U" (Jaccard-type) and "C" (Sorensen-type; one minus the Morisita-Horn similarity at q = 2) are the corresponding overlap complements (proportions of non-shared units). At q = 0, "U" equals "S" and "C" equals "V"; at q = 1, "U" equals "C".
tree	A phylogenetic tree of class phylo whose tip labels match the taxa in data.
dist	A functional distance matrix (or dist) over the taxa.
tau	Optional functional distance threshold. Defaults to max(dist).
type	Diversity type: "auto" (default) infers it from the inputs (counts only -> neutral, +tree -> phylogenetic, +dist -> functional); an explicit "neutral", "phylogenetic" or "functional" asserts the type and is validated against the inputs (e.g. "phylogenetic" requires a tree; "neutral" ignores any tree/dist carried by the object).
by	"pairwise" (default) compares every pair of samples; "collective" compares all samples together or each specified group.
groups	For by = "collective", an optional named list of character vectors of sample names. Each group must contain at least two distinct samples. Defaults to one group named "all" containing every sample.
sets	An optional named list of character vectors of taxon names, as in hillshare(). Defaults to one set per taxon. Sets may overlap.

Details

For each taxon (or phylogenetic branch), the contribution kernel is the Jensen gap between its pooled abundance and its abundances across samples. The gap is zero if its abundance is constant. At q = 0 the gap measures uneven occurrence; at q = 1 it is the entropy gap; at other orders it is the power-sum gap. For neutral and functional diversity the kernel belongs directly to a taxon. For phylogenetic diversity each branch's gap is assigned to descendant taxa in proportion to their own neutral Jensen gaps at the same order. This is one explicit allocation of shared branches, rather than a unique taxon-level decomposition of phylogenetic turnover.

The kernel fractions are multiplied by the selected dissimilarity metric. Thus singleton-taxon contributions add exactly to that metric, while their relative ranking is the same for S, C, U, and V at a given order and comparison. Zero-length branches and taxa absent from every selected sample receive zero. Every selected sample must have a positive total. Functional beta uses the package's raw-count partitioning, so unequal sample totals can contribute to functional dissimilarity even when relative composition is unchanged.

Value

A data frame with columns first, second, group, q, type, metric, set, contribution, share, and rank. Pairwise rows use first and second; collective rows use group. Rows are ranked from

largest to smallest contribution within each comparison, order, and metric. Sets that omit taxa sum to less than the total dissimilarity; overlapping sets can sum to more.

See Also

`hillshare()`, `hillpair()`, `hilldiss()`

Examples

```
counts <- matrix(c(5, 0, 2, 3, 4, 0, 0, 2, 6), nrow = 3,
                 dimnames = list(c("A", "B", "C"), c("s1", "s2", "s3")))
hillcontrib(counts, q = 1, metric = "C")
hillcontrib(counts, q = 1, metric = "C", by = "collective")
```

`hilldiss` *Hill numbers-based dissimilarity*

Description

Compute overall (multi-sample) dissimilarity metrics from the Hill-number beta diversity following Chiu et al. (2014). These are the complements of the similarities returned by `hillsim()`. With `hierarchy`, each transition's beta is converted using its data-dependent effective ceiling rather than the number of child units, keeping the metrics in `[0, 1]`.

Usage

```
hilldiss(
  data,
  q = c(0, 1, 2),
  metric = c("S", "C", "U", "V"),
  tree = NULL,
  dist = NULL,
  tau = NULL,
  type = c("auto", "neutral", "phylogenetic", "functional"),
  out = c("tibble", "matrix"),
  hierarchy = NULL,
  metadata = NULL,
  weights = "equal",
  depth = NULL
)
```

Arguments

`data`	A count table (taxa x samples) or a supported object; a single sample is not meaningful for partitioning.
`q`	Numeric vector of diversity orders (>= 0). Defaults to `c(0, 1, 2)` (richness, Shannon, Simpson).

`metric` Dissimilarity metric(s) to return, any of `"S"`, `"C"`, `"U"`, `"V"`. Defaults to all four. `"S"` (Jaccard-type) and `"V"` (Sorensen-type) are turnover measures, relative to the pooled assemblage and to a single assemblage respectively; `"U"` (Jaccard-type) and `"C"` (Sorensen-type; one minus the Morisita-Horn similarity at `q = 2`) are the corresponding overlap complements (proportions of non-shared units). At `q = 0`, `"U"` equals `"S"` and `"C"` equals `"V"`; at `q = 1`, `"U"` equals `"C"`.

`tree` A phylogenetic tree of class `phylo` whose tip labels match the taxa in `data`.

`dist` A functional distance matrix (or `dist`) over the taxa.

`tau` Optional functional distance threshold. Defaults to `max(dist)`.

`type` Diversity type: `"auto"` (default) infers it from the inputs (counts only -> neutral, `+tree` -> phylogenetic, `+dist` -> functional); an explicit `"neutral"`, `"phylogenetic"` or `"functional"` asserts the type and is validated against the inputs (e.g. `"phylogenetic"` requires a `tree`; `"neutral"` ignores any tree/dist carried by the object).

`out` Output shape: `"tibble"` (default) returns a long-format `data.frame` with columns `q`, `metric`, `value`; `"matrix"` returns the legacy matrix (orders in rows, metrics in columns, dropped to a vector for a single metric). With `hierarchy`, the table adds a `scale` column for each transition above the sample scale. A `tau` grid adds a `tau` column. Matrix output has one row per `q` and transition, with a named list of matrices for a `tau` grid.

`hierarchy` Optional one-sided nesting formula, coarsest to finest, e.g. `~ region / site`. Every transition above `sample`, including `total`, gets its own dissimilarity metrics. Each run uses one shared tree `depth` or functional `tau`; a functional `tau` grid runs the full hierarchy once per threshold.

`metadata` Optional per-sample `data.frame` supplying the variables named in `hierarchy`; rows are matched to the count-table columns by name when possible, otherwise by position.

`weights` Sample weights for hierarchical partitioning: `"equal"` (default), `"abundance"` (proportional to each sample's total count), or a positive numeric vector with one value per sample. A named vector is matched to count-table columns; an unnamed vector uses column order. Weights are normalised to sum to one. Requires `hierarchy`.

`depth` Optional positive phylogenetic reference depth for hierarchical partitioning. On an ultrametric tree, use only branch segments within the last `depth` units before the tips. Must not exceed the tree height. `NULL` (default) uses the current shared mean-depth normaliser. Requires `hierarchy` and a phylogenetic `tree`.

Value

A long-format `data.frame` of class `hill_dissimilarity` (default, with a `plot()` method), or a matrix/vector of dissimilarities when `out = "matrix"`.

See Also

`hillsim()`, `hillpair()`, `hillpart()`

Examples

```
counts <- matrix(c(10, 0, 5, 2, 8, 1), nrow = 3,
                 dimnames = list(c("t1", "t2", "t3"), c("s1", "s2")))
hilldiss(counts)
plot(hilldiss(counts))
```

`hilldiv` *Hill numbers computation*

Description

Compute neutral, phylogenetic and/or functional Hill numbers (alpha diversity) from a single sample or a count table. By default the computation is *cumulative*: every diversity type whose inputs are present is returned. Counts are always available, so neutral is always computed; a `tree` adds phylogenetic and a `dist` adds functional. Supplying both a `tree` and a `dist` therefore returns neutral, phylogenetic and functional side by side in a single tibble (with a `type` column). Use `type` to restrict the output to a subset.

Usage

```
hilldiv(
  data,
  q = c(0, 1, 2),
  tree = NULL,
  dist = NULL,
  tau = NULL,
  type = c("auto", "neutral", "phylogenetic", "functional"),
  reference = c("pool", "sample"),
  out = c("tibble", "matrix")
)
```

Arguments

`data`	Counts: a numeric vector (one sample), a matrix/data.frame (taxa x samples), a `phyloseq` object or a `TreeSummarizedExperiment`.
`q`	Numeric vector of diversity orders (>= 0). Defaults to `c(0, 1, 2)` (richness, Shannon, Simpson).
`tree`	A phylogenetic tree of class `phylo` whose tip labels match the taxa in `data`.
`dist`	A functional distance matrix (or `dist`) over the taxa.
`tau`	Optional functional distance threshold. Defaults to `max(dist)`.
`type`	Diversity type(s) to compute. `"auto"` (default) returns every type whose inputs are present (always neutral, plus phylogenetic with a `tree` and functional with a `dist`). Pass an explicit type, or a character vector of types, to restrict the output – e.g. `"neutral"` ignores any tree/dist carried by the object, and `c("neutral", "phylogenetic")` drops functional even when a `dist` is supplied. A requested type that lacks its input (e.g. `"phylogenetic"` without a `tree`) is an error.

`reference` Reference tree depth for *phylogenetic* Hill numbers (ignored for neutral and functional types). `"pool"` (default) reads every sample at one common depth `T = mean(T_j)`, so values share a comparable axis across samples; `"sample"` reads each sample at its own depth `T_j` (effective lineages at that sample's depth). The two coincide on ultrametric trees. This reference depth is intentionally *not* offered by `hillpart()`: in a partition `T` is fixed at the mean per-sample depth of Chiu et al. (2014), the unique value for which `gamma / alpha` is a valid decomposition with `beta` in `[1, N]`.

`out` Output shape: `"tibble"` (default) returns a long-format `data.frame` with columns `q`, `sample`, `value` (plus a `type` column when more than one type is computed) and `print()`/`plot()` methods; `"matrix"` returns a matrix with samples in rows and diversity orders (`q0`, `q1`, ...) in columns, or, when more than one type is computed, a named list of such matrices (one per type).

Value

A long-format `data.frame` of class `hill_diversity` (default). With `out = "matrix"`, a matrix of Hill numbers (samples in rows, diversity orders `q0`, `q1`, ... in columns) for a single type, or a named list of such matrices when several types are computed.

References

Chao, A., Chiu, C.-H. & Jost, L. (2010). Phylogenetic diversity measures based on Hill numbers. Phil. Trans. R. Soc. B, 365, 3599-3609.

Alberdi, A. & Gilbert, M.T.P. (2019). A guide to the application of Hill numbers to DNA-based diversity analyses. Mol. Ecol. Resour., 19, 804-817.

See Also

`hillpart()`, `hilldiss()`, `hillprof()`

Examples

```
counts <- matrix(c(10, 0, 5, 2, 8, 1), nrow = 3,
                 dimnames = list(c("t1", "t2", "t3"), c("s1", "s2")))
hilldiv(counts)
hilldiv(counts, q = c(0, 1, 2))
plot(hilldiv(counts, q = c(0, 1, 2)))

# Supplying both a tree and a distance matrix returns neutral, phylogenetic
# and functional diversity together, distinguished by a `type` column.
tree <- ape::read.tree(text = "((t1:1,t2:1):1,t3:2);")
dist <- as.matrix(stats::dist(c(t1 = 0, t2 = 1, t3 = 4)))
hilldiv(counts, tree = tree, dist = dist)

# Restrict the output with `type` (a scalar or a vector):
hilldiv(counts, tree = tree, dist = dist, type = c("neutral", "functional"))
```

`hilleven` *Hill-number evenness*

Description

Evenness expressed through Hill numbers as the ratio of diversity of order `q` to richness (`qD / 0D`), which ranges from 0 to 1.

Usage

```
hilleven(
  data,
  q = c(1, 2),
  tree = NULL,
  dist = NULL,
  tau = NULL,
  type = c("auto", "neutral", "phylogenetic", "functional"),
  out = c("tibble", "matrix")
)
```

Arguments

`data`	Counts: a numeric vector (one sample), a matrix/data.frame (taxa x samples), a `phyloseq` object or a `TreeSummarizedExperiment`.
`q`	Numeric vector of diversity orders (> 0 are meaningful for evenness). Defaults to `c(1, 2)`.
`tree`	A phylogenetic tree of class `phylo` whose tip labels match the taxa in `data`.
`dist`	A functional distance matrix (or `dist`) over the taxa.
`tau`	Optional functional distance threshold. Defaults to `max(dist)`.
`type`	Diversity type: `"auto"` (default) infers it from the inputs (counts only -> neutral, `+tree` -> phylogenetic, `+dist` -> functional); an explicit `"neutral"`, `"phylogenetic"` or `"functional"` asserts the type and is validated against the inputs (e.g. `"phylogenetic"` requires a `tree`; `"neutral"` ignores any tree/dist carried by the object).
`out`	Output shape: `"tibble"` (default) returns a long-format `data.frame` with columns `q`, `sample`, `value`; `"matrix"` returns a matrix with samples in rows and orders in columns.

Value

A long-format `data.frame` of class `hill_evenness` (default) with a `plot()` method, or a matrix of evenness values (samples in rows, orders in columns) when `out = "matrix"`.

See Also

`hilldiv()`

Examples

```
counts <- matrix(c(10, 0, 5, 2, 8, 1), nrow = 3,
                 dimnames = list(c("t1", "t2", "t3"), c("s1", "s2")))
hilleven(counts)
plot(hilleven(counts, q = c(1, 1.5, 2)))
```

`hillpair` *Pairwise Hill numbers-based dissimilarity*

Description

Compute dissimilarity metrics for every pair of samples, returning distance objects suitable for ordination (e.g. NMDS, PCoA). With `hierarchy`, compare units at `level`: each unit pools the relative abundances of its samples using their normalised sample weights. Abundance weights make this equivalent to pooling the original counts.

Usage

```
hillpair(
  data,
  q = c(0, 1, 2),
  metric = c("S", "C", "U", "V"),
  tree = NULL,
  dist = NULL,
  tau = NULL,
  type = c("auto", "neutral", "phylogenetic", "functional"),
  out = c("dist", "tibble"),
  parallel = FALSE,
  hierarchy = NULL,
  metadata = NULL,
  weights = "equal",
  level = NULL
)
```

Arguments

`data`	A count table (taxa x samples) or a supported object; a single sample is not meaningful for partitioning.
`q`	Numeric vector of diversity orders (>= 0). Defaults to `c(0, 1, 2)` (richness, Shannon, Simpson).
`metric`	Dissimilarity metric(s) to return, any of `"S"`, `"C"`, `"U"`, `"V"`. Defaults to all four. `"S"` (Jaccard-type) and `"V"` (Sorensen-type) are turnover measures, relative to the pooled assemblage and to a single assemblage respectively; `"U"` (Jaccard-type) and `"C"` (Sorensen-type; one minus the Morisita-Horn similarity at `q = 2`) are the corresponding overlap complements (proportions of non-shared units). At `q = 0`, `"U"` equals `"S"` and `"C"` equals `"V"`; at `q = 1`, `"U"` equals `"C"`.

tree	A phylogenetic tree of class phylo whose tip labels match the taxa in data.
dist	A functional distance matrix (or dist) over the taxa.
tau	Optional functional distance threshold. Defaults to max(dist).
type	Diversity type: "auto" (default) infers it from the inputs (counts only -> neutral, +tree -> phylogenetic, +dist -> functional); an explicit "neutral", "phylogenetic" or "functional" asserts the type and is validated against the inputs (e.g. "phylogenetic" requires a tree; "neutral" ignores any tree/dist carried by the object).
out	Output type: "dist" (default) returns a dist object per requested metric/order combination; "tibble" returns a long-format table.
parallel	Logical; if TRUE and furrr is installed, compute pairs in parallel.
hierarchy	Optional nesting formula. When supplied, compare units at level after pooling their samples with weights.
metadata	Optional per-sample data.frame supplying the variables named in hierarchy; rows are matched to the count-table columns by name when possible, otherwise by position.
weights	Sample weights for hierarchical partitioning: "equal" (default), "abundance" (proportional to each sample's total count), or a positive numeric vector with one value per sample. A named vector is matched to count-table columns; an unnamed vector uses column order. Weights are normalised to sum to one. Requires hierarchy.
level	Name of the hierarchy level to compare. Defaults to the finest named level in hierarchy.

Details

The type-specific structure (per-sample normalisation, the tree traversal or the functional similarity product) is computed **once** over all samples via the partitioning engine; each pair then only combines its two precomputed columns into beta, which is turned into the requested dissimilarity metrics. The maths are therefore identical to hilldiss() on two samples, without re-running the full engine per pair. When parallel = TRUE and the furrr package is installed, pairs are computed in parallel via the active future plan. A progressr progress bar is reported when that package is installed and a handler is active.

Value

For out = "dist", a named list of dist objects (one per order/metric, named e.g. "q0S"), collapsed to a single dist when only one combination is requested. For out = "tibble", a long-format data.frame with columns first, second, q, metric, value. Hierarchical output labels units at the selected level in place of samples.

See Also

hilldiss(), hilldiv()

Examples

```
counts <- matrix(c(10, 0, 5, 2, 8, 1, 3, 4, 0, 6, 2, 7), nrow = 3,
                 dimnames = list(c("t1", "t2", "t3"),
                                 c("s1", "s2", "s3", "s4")))
hillpair(counts, q = 1, metric = "C")
```

`hillpart` *Hill numbers diversity partitioning*

Description

Partition neutral, phylogenetic or functional Hill-number diversity into alpha, gamma and beta components across a set of samples. With a `hierarchy` formula it instead performs *multi-scale* (nested) partitioning, returning one beta per hierarchical level.

Usage

```
hillpart(
  data,
  q = c(0, 1, 2),
  tree = NULL,
  dist = NULL,
  tau = NULL,
  hierarchy = NULL,
  metadata = NULL,
  type = c("auto", "neutral", "phylogenetic", "functional"),
  out = c("tibble", "matrix"),
  weights = "equal",
  depth = NULL
)
```

Arguments

`data`	A count table (taxa x samples) or a supported object; a single sample is not meaningful for partitioning.
`q`	Numeric vector of diversity orders (>= 0). Defaults to `c(0, 1, 2)` (richness, Shannon, Simpson).
`tree`	A phylogenetic tree of class `phylo` whose tip labels match the taxa in `data`.
`dist`	A functional distance matrix (or `dist`) over the taxa.
`tau`	Optional functional distance threshold. Defaults to `max(dist)`.
`hierarchy`	Optional one-sided nesting formula, coarsest to finest, e.g. `~ region / site`, requesting multi-scale (nested) partitioning instead of the default single-level partition. One beta is returned per hierarchical transition and the chain telescopes exactly: `gamma = alpha_finest * prod(beta)`. Works for all three diversity types (neutral, phylogenetic, functional); see the partitioning vignette for the

shared construction. A single tree `depth` or functional `tau` is used across every scale in a partition, so `alpha * prod(beta) = gamma` exactly and every beta is at least one. Grouping variables are resolved against `metadata` when supplied, otherwise against the calling environment. Linear unit weighting changes results relative to the older `p_u^q` formula in unbalanced designs for `q != 1` (a few per cent in neutral examples; balanced equal-weight designs are identical). With functional diversity it can also shift `q = 1` slightly (about 0.003 in the gut data), because a unit's measure-weighted mass need not equal its sample weight. For functional diversity, a vector of `tau` values runs the full hierarchy once per threshold; per-scale thresholds are not supported.

`metadata` Optional per-sample `data.frame` supplying the variables named in `hierarchy`; rows are matched to the count-table columns by name when possible, otherwise by position.

`type` Diversity type: `"auto"` (default) infers it from the inputs (counts only -> neutral, `+tree` -> phylogenetic, `+dist` -> functional); an explicit `"neutral"`, `"phylogenetic"` or `"functional"` asserts the type and is validated against the inputs (e.g. `"phylogenetic"` requires a `tree`; `"neutral"` ignores any tree/dist carried by the object).

`out` Output shape: `"tibble"` (default) returns a long-format `data.frame` with columns `q`, `component`, `value`; `"matrix"` returns the legacy matrix (orders in rows, `alpha`/`gamma`/`beta` in columns). With `hierarchy`, `"tibble"` returns one row per `(q, scale)` and `"matrix"` returns `alpha`, one `beta_<level>` per nesting level, and `gamma`. A multi-`tau` hierarchy returns an extra `tau` column for `"tibble"`, or a named list of matrices for `"matrix"`.

`weights` Sample weights for hierarchical partitioning: `"equal"` (default), `"abundance"` (proportional to each sample's total count), or a positive numeric vector with one value per sample. A named vector is matched to count-table columns; an unnamed vector uses column order. Weights are normalised to sum to one. Requires `hierarchy`.

`depth` Optional positive phylogenetic reference depth for hierarchical partitioning. On an ultrametric tree, use only branch segments within the last `depth` units before the tips. Must not exceed the tree height. `NULL` (default) uses the current shared mean-depth normaliser. Requires `hierarchy` and a phylogenetic `tree`.

Value

A long-format `data.frame` of class `hill_partition` (default) with a `plot()` method, or a matrix with columns `alpha`, `gamma`, `beta` and diversity orders in rows when `out = "matrix"`. With `hierarchy`, a `hill_hierarchy` long-format `data.frame` (with its own `plot()` method) or the corresponding wide matrix.

See Also

`hilldiv()`, `hilldiss()`, `hillsim()`

Examples

```
counts <- matrix(c(10, 0, 5, 2, 8, 1), nrow = 3,
                 dimnames = list(c("t1", "t2", "t3"), c("s1", "s2")))
```

```
hillpart(counts)
plot(hillpart(counts))

# Multi-scale partitioning across a nested design.
set.seed(1)
tab <- matrix(rpois(12 * 8, 5), nrow = 12,
              dimnames = list(paste0("t", 1:12), paste0("s", 1:8)))
md <- data.frame(region = rep(c("N", "S"), each = 4),
                 site = rep(c("a", "b", "c", "d"), each = 2),
                 row.names = paste0("s", 1:8))
hillpart(tab, hierarchy = ~ region / site, metadata = md)
```

`hillprof` *Diversity profile across a range of orders*

Description

Compute a diversity profile: Hill numbers evaluated over a fine sweep of diversity orders `q`. Profiles are the standard diagnostic for comparing the diversity of assemblages, since the ranking of samples can change with `q`.

Usage

```
hillprof(
  data,
  q = seq(0, 3, by = 0.1),
  tree = NULL,
  dist = NULL,
  tau = NULL,
  type = c("auto", "neutral", "phylogenetic", "functional"),
  reference = c("pool", "sample"),
  out = c("tibble", "matrix")
)
```

Arguments

`data`	Counts: a numeric vector (one sample), a matrix/data.frame (taxa x samples), a `phyloseq` object or a `TreeSummarizedExperiment`.
`q`	Numeric vector of diversity orders to evaluate. Defaults to a fine sweep from 0 to 3.
`tree`	A phylogenetic tree of class `phylo` whose tip labels match the taxa in `data`.
`dist`	A functional distance matrix (or `dist`) over the taxa.
`tau`	Optional functional distance threshold. Defaults to `max(dist)`.
`type`	Diversity type: `"auto"` (default) infers it from the inputs (counts only -> neutral, `+tree` -> phylogenetic, `+dist` -> functional); an explicit `"neutral"`, `"phylogenetic"` or `"functional"` asserts the type and is validated against the inputs (e.g. `"phylogenetic"` requires a `tree`; `"neutral"` ignores any tree/dist carried by the object).

`reference` Reference tree depth for *phylogenetic* Hill numbers (ignored for neutral and functional types). `"pool"` (default) reads every sample at one common depth `T = mean(T_j)`, so values share a comparable axis across samples; `"sample"` reads each sample at its own depth `T_j` (effective lineages at that sample's depth). The two coincide on ultrametric trees. This reference depth is intentionally *not* offered by `hillpart()`: in a partition `T` is fixed at the mean per-sample depth of Chiu et al. (2014), the unique value for which `gamma / alpha` is a valid decomposition with `beta` in `[1, N]`.

`out` Output type: `"tibble"` (default, long format ready for plotting) or `"matrix"`.

Value

A long-format `data.frame` of class `hill_profile` (columns `q`, `sample`, `value`) with a plot() method, or a matrix (samples in rows, orders in columns) when `out = "matrix"`.

See Also

`hilldiv()`

Examples

```
counts <- matrix(c(10, 0, 5, 2, 8, 1), nrow = 3,
                 dimnames = list(c("t1", "t2", "t3"), c("s1", "s2")))
prof <- hillprof(counts)
plot(prof)
```

---

hillred *Hill numbers redundancy*

Description

Estimate phylogenetic or functional redundancy by fitting the saturating relationship between neutral diversity and phylogenetic/functional diversity across samples: `y = -a * 2^(-x / b) + c`. Redundancy is summarised as `1 - b / max(x)`.

Usage

```
hillred(
  data,
  q = c(0, 1, 2),
  tree = NULL,
  dist = NULL,
  tau = NULL,
  type = c("auto", "phylogenetic", "functional"),
  reference = c("pool", "sample"),
  out = c("tibble", "matrix")
)
```

Arguments

data	A count table (taxa x samples); requires either tree or dist.
q	Numeric vector of diversity orders (>= 0). Defaults to c(0, 1, 2) (richness, Shannon, Simpson).
tree	A phylogenetic tree of class phylo whose tip labels match the taxa in data.
dist	A functional distance matrix (or dist) over the taxa.
tau	Optional functional distance threshold. Defaults to max(dist).
type	Diversity type: "auto" (default) infers it from the inputs (counts only -> neutral, +tree -> phylogenetic, +dist -> functional); an explicit "neutral", "phylogenetic" or "functional" asserts the type and is validated against the inputs (e.g. "phylogenetic" requires a tree; "neutral" ignores any tree/dist carried by the object).
reference	Reference tree depth for *phylogenetic* Hill numbers (ignored for neutral and functional types). "pool" (default) reads every sample at one common depth T = mean(T_j), so values share a comparable axis across samples; "sample" reads each sample at its own depth T_j (effective lineages at that sample's depth). The two coincide on ultrametric trees. This reference depth is intentionally *not* offered by hillpart(): in a partition T is fixed at the mean per-sample depth of Chiu et al. (2014), the unique value for which gamma / alpha is a valid decomposition with beta in [1, N].
out	Output shape: "tibble" (default) returns a data.frame with one row per q and columns q, redundancy, a, b, c; "matrix" returns the legacy matrix (orders in rows).

Value

A data.frame of class hill_redundancy (default) with a plot() method, or a matrix with columns redundancy, a, b, c (one row per q) when out = "matrix". The tibble carries the per-sample neutral and phylogenetic/functional diversity used for the fit as a "hill_fit" attribute, which the plot method draws.

See Also

hilldiv(), plot.hill_redundancy()

Examples

```
d <- traits2dist(gut_traits)
red <- hillred(gut_counts, dist = d)
red
plot(red)
```

`hillshare` *Allocate Hill power-sum weight to taxa or sets*

Description

Calculate the share of a community's neutral, phylogenetic, or functional Hill power sum assigned to each taxon or a specified set of taxa. Shares range from zero to one within a nonempty sample. They describe an additive allocation of the power sum, not a percentage of the effective Hill number.

Usage

```
hillshare(
  data,
  q = c(0, 1, 2),
  tree = NULL,
  dist = NULL,
  tau = NULL,
  type = c("auto", "neutral", "phylogenetic", "functional"),
  sets = NULL,
  allocation = "qpower"
)
```

Arguments

`data`	Counts: a numeric vector or a taxa-by-samples matrix/data frame. The input adapters accepted by `hilldiv()` are also supported.
`q`	Nonnegative diversity orders. Defaults to c(0, 1, 2).
`tree`	Optional rooted phylogenetic tree of class phylo.
`dist`	Optional functional distance matrix over the taxa.
`tau`	Functional distance threshold. Defaults to max(dist).
`type`	Diversity type(s). "auto" (default) computes neutral shares and also phylogenetic and functional shares when their inputs are supplied. Explicit types restrict the output as in `hilldiv()`.
`sets`	A named list whose elements are character vectors of taxon names. Each element can contain one taxon or several. By default, every taxon is returned as a singleton set. Sets may overlap, and each set's share is the sum of its members' allocations. An unnamed character vector is accepted as a request for singleton taxa.
`allocation`	Allocation rule. The first iteration supports "qpower".

Details

Let p_i be the relative abundance of taxon i. Neutral weight is p_i^q, normalised by its sum over present taxa.

For a phylogenetic branch b with length L_b and descendant abundance a_b, branch weight is L_b a_b^q. That weight is divided among its present descendants in proportion to p_i^q. At q = 0, present descendants share the branch equally. Branch weights and taxon allocations are normalised by the sum of all branch weights.

For functional diversity, similarity is s_ij = 1 - min(d_ij, tau) / tau and a_i = sum_j s_ij p_j. The existing functional Hill power sum in hilldiv3 is sum_i p_i a_i^(q-1), so taxon i receives p_i a_i^(q-1) / sum_j p_j a_j^(q-1). With identity similarity this reduces to the neutral rule. At q = 1, functional shares equal p_i, although the effective functional Hill number still depends on similarities.

An empty sample has no power sum to allocate, so its shares are NA.

Value

A data frame with columns q, sample, type, set, share, and allocation. Shares for distinct singleton taxa sum to one in each nonempty sample, diversity type, and order.

References

Chao, A., Chiu, C.-H. & Jost, L. (2010). Phylogenetic diversity measures based on Hill numbers. Phil. Trans. R. Soc. B, 365, 3599-3609.

Leinster, T. & Cobbold, C.A. (2012). Measuring diversity: the importance of species similarity. Ecology, 93, 477-489.

See Also

`hilldiv()`, `hillprof()`

Examples

```
counts <- c(A = 394, B = 202, C = 354, D = 50)
tree <- ape::read.tree(
  text = "(((A:0.300,B:0.300):0.185,C:0.485):0.009,D:0.494);"
)
hillshare(counts, q = 3, tree = tree,
          sets = list(C = "C", ABC = c("A", "B", "C")))
d <- as.matrix(stats::dist(c(A = 0, B = 0.5, C = 1, D = 2)))
hillshare(counts, q = c(0, 1, 2), dist = d, type = "functional",
          sets = list(C = "C", ABC = c("A", "B", "C")))
```

`hillsim` *Hill numbers-based similarity*

Description

Compute overall similarity metrics from the Hill-number beta diversity (Chiu et al. 2014). These are `1 -` the dissimilarities from `hilldiss()`.

Usage

```
hillsim(
  data,
  q = c(0, 1, 2),
  metric = c("S", "C", "U", "V"),
  tree = NULL,
  dist = NULL,
  tau = NULL,
  type = c("auto", "neutral", "phylogenetic", "functional"),
  out = c("tibble", "matrix"),
  hierarchy = NULL,
  metadata = NULL,
  weights = "equal",
  depth = NULL
)
```

Arguments

data	A count table (taxa x samples) or a supported object; a single sample is not meaningful for partitioning.
q	Numeric vector of diversity orders (>= 0). Defaults to c(0, 1, 2) (richness, Shannon, Simpson).
metric	Similarity metric(s) to return, any of "S", "C", "U", "V". Defaults to all four. Each is one minus the dissimilarity of the same code in hilldiss(), so identical assemblages give 1. "C" at q = 2 is the Morisita-Horn similarity.
tree	A phylogenetic tree of class phylo whose tip labels match the taxa in data.
dist	A functional distance matrix (or dist) over the taxa.
tau	Optional functional distance threshold. Defaults to max(dist).
type	Diversity type: "auto" (default) infers it from the inputs (counts only -> neutral, +tree -> phylogenetic, +dist -> functional); an explicit "neutral", "phylogenetic" or "functional" asserts the type and is validated against the inputs (e.g. "phylogenetic" requires a tree; "neutral" ignores any tree/dist carried by the object).
out	Output shape: "tibble" (default) returns a long-format data.frame with columns q, metric, value; "matrix" returns the legacy matrix (orders in rows, metrics in columns, dropped to a vector for a single metric). With hierarchy, the table adds a scale column for each transition above the sample scale. A tau grid adds a tau column. Matrix output has one row per q and transition, with a named list of matrices for a tau grid.
hierarchy	Optional one-sided nesting formula, coarsest to finest, e.g. ~ region / site. Every transition above sample, including total, gets its own dissimilarity metrics. Each run uses one shared tree depth or functional tau; a functional tau grid runs the full hierarchy once per threshold.
metadata	Optional per-sample data.frame supplying the variables named in hierarchy; rows are matched to the count-table columns by name when possible, otherwise by position.

weights	Sample weights for hierarchical partitioning: "equal" (default), "abundance" (proportional to each sample's total count), or a positive numeric vector with one value per sample. A named vector is matched to count-table columns; an unnamed vector uses column order. Weights are normalised to sum to one. Requires hierarchy.
depth	Optional positive phylogenetic reference depth for hierarchical partitioning. On an ultrametric tree, use only branch segments within the last depth units before the tips. Must not exceed the tree height. NULL (default) uses the current shared mean-depth normaliser. Requires hierarchy and a phylogenetic tree.

Value

A long-format data.frame of class hill_similarity (default, with a plot() method), or a matrix/vector of similarities when out = "matrix".

See Also

hilldiss()

Examples

```
counts <- matrix(c(10, 0, 5, 2, 8, 1), nrow = 3,
                 dimnames = list(c("t1", "t2", "t3"), c("s1", "s2")))
hillsim(counts)
plot(hillsim(counts))
```

---

match_data *Match and align a count table to a tree or distance matrix*

Description

Subsets and reorders a count table so that its taxa match those of a phylogenetic tree or a functional distance matrix, dropping taxa absent from the reference. This realises the match_data() helper that hilldiv2's documentation referred to but never provided.

Usage

```
match_data(data, tree = NULL, dist = NULL)
```

Arguments

data	A count matrix/data.frame (taxa x samples) with row names.
tree	A phylo tree (optional).
dist	A distance matrix (optional).

Value

The count matrix restricted to and ordered by the shared taxa.

Examples

```
counts <- matrix(1:6, nrow = 3,
                 dimnames = list(c("t1", "t2", "t3"), c("s1", "s2")))
tree <- ape::read.tree(text = "((t1:1,t2:1):1,t4:2);")
match_data(counts, tree = tree)
```

---

plot.hill_profile *Plot a diversity profile*

Description

Base-graphics plot of a `hillprof()` result: one line per sample showing the Hill number against the diversity order q.

Usage

```
## S3 method for class 'hill_profile'
plot(x, ...)
```

Arguments

x	A `hill_profile` object from `hillprof()`.
...	Further arguments passed to `plot()`.

Value

The `hill_profile` object, invisibly.

---

plot.hill_redundancy *Plot a redundancy fit*

Description

Base-graphics plot of a `hillred()` result. For each diversity order q it shows the per-sample neutral diversity (x) against phylogenetic/functional diversity (y), overlaid with the fitted saturating curve `y = -a * 2^(-x / b) + c`. A curve that bends sharply and plateaus well below the points' spread indicates high redundancy; a near-linear fit indicates low redundancy. This mirrors the profile plot of `hillprof()`.

Usage

```
## S3 method for class 'hill_redundancy'
plot(x, ...)
```

Arguments

x	A hill_redundancy object from hillred().
...	Further arguments passed to plot().

Value

The hill_redundancy object, invisibly.

See Also

hillred(), plot.hill_profile()

---

traits2dist *Convert a trait table into a distance matrix*

Description

Build a pairwise functional distance matrix from a table of taxon traits, suitable as the dist argument of hilldiv() and friends.

Usage

```
traits2dist(traits, method = c("gower", "euclidean", "manhattan"))
```

Arguments

traits	A table with taxa (OTUs/ASVs/MAGs) in rows and traits in columns. Traits may be continuous, binary or proportional.
method	Distance metric passed to cluster::daisy(): "gower" (default), "euclidean" or "manhattan".

Value

A numeric distance matrix.

Examples

```
traits <- data.frame(body = c(1, 0.2, 0.9), diet = c(0L, 1L, 1L),
                     row.names = c("t1", "t2", "t3"))
traits2dist(traits)
```

`tss` *Total Sum Scaling normalisation*

Description

Normalise a numeric vector or count matrix so that each sample (column) sums to one. Columns that sum to zero are returned as all-zero (the `0/0` = `NaN` case is mapped to `0`).

Usage

```
tss(abund)
```

Arguments

abund	A numeric vector or a matrix/data.frame of counts with taxa (OTUs/ASVs/MAGs) in rows and samples in columns.

Value

A normalised object of the same shape as abund (vector in, vector out; matrix/data.frame in, matrix out).

Examples

```
tss(c(a = 1, b = 3))
tss(matrix(c(1, 0, 3, 0, 0, 2), nrow = 3))
```

Index

∗ **datasets**
gut_counts, 2
gut_traits, 3
gut_tree, 4

cluster::daisy(), *23*

gut_counts, 2, *3, 4*
gut_traits, *3*, 3, *4*
gut_tree, *3*, 4

hillcontrib, 4
hilldiss, 6
hilldiss(), *4*, *6*, *9*, *12*, *14*, *19–21*
hilldiv, 8
hilldiv(), *10*, *12*, *14*, *16–19*, *23*
hilleven, 10
hillpair, 11
hillpair(), *4*, *6, 7*
hillpart, 13
hillpart(), *7*, *9*, *16, 17*
hillprof, 15
hillprof(), *9*, *19*, *22*
hillred, 16
hillred(), *22, 23*
hillshare, 18
hillshare(), *5, 6*
hillsim, 19
hillsim(), *6, 7*, *14*

match_data, 21

plot(), *16, 17*, *22, 23*
plot.hill_profile, 22
plot.hill_profile(), *23*
plot.hill_redundancy, 22
plot.hill_redundancy(), *17*

traits2dist, 23
traits2dist(), *3*
tss, 24