Package: ecodive (via r-universe)

June 3, 2026

Type Package

Title Parallel and Memory-Efficient Ecological Diversity Metrics

Version 2.3.0

Description Computes alpha and beta diversity metrics using concurrent 'C' threads. Metrics include 'UniFrac', Faith's phylogenetic diversity, Bray-Curtis dissimilarity, Shannon diversity index, and many others. Also parses newick trees into 'phylo' objects and rarefies feature tables. Parallel and memory-efficient algorithms are described in Smith et al. (2026) <doi:10.21105/joss.09777>.

URL https://cmmr.github.io/ecodive/, https://github.com/cmmr/ecodive

BugReports https://github.com/cmmr/ecodive/issues

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Depends R (>= 3.6.0)

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Config/Needs/website rmarkdown

Config/testthat/edition 3

Imports parallel, utils

Suggests knitr, Matrix, parallelly, rmarkdown, slam, tinytest

VignetteBuilder knitr

Repository https://cmmr.r-universe.dev

Date/Publication 2026-06-03 22:20:32 UTC

RemoteUrl https://github.com/cmmr/ecodive

RemoteRef HEAD

RemoteSha e70f237f61153a971f72e3a083623f487e4ad665

Contents

ace . . . 3
aitchison . . . 5
alpha_div . . . 7
berger . . . 9
beta_div . . . 10
bhattacharyya . . . 14
bray . . . 15
brillouin . . . 18
canberra . . . 19
chao1 . . . 21
chebyshev . . . 22
chord . . . 24
clark . . . 26
divergence . . . 28
euclidean . . . 30
ex_counts . . . 32
ex_tree . . . 33
faith . . . 34
fisher . . . 35
generalized_unifrac . . . 36
gower . . . 38
hamming . . . 40
hellinger . . . 42
horn . . . 43
inv_simpson . . . 45
jaccard . . . 47
jensen . . . 48
jsd . . . 50
list_metrics . . . 51
lorentzian . . . 54
manhattan . . . 56
margalef . . . 58
matusita . . . 59
mcintosh . . . 61
menhinick . . . 62
minkowski . . . 63
morisita . . . 66
motyka . . . 67
n_cpus . . . 69
normalized_unifrac . . . 70
observed . . . 72
ochiai . . . 73
psym_chisq . . . 74
rarefy . . . 76
read_tree . . . 78
robust_aitchison . . . 79

shannon 80
simpson 82
soergel 83
sorensen 85
squared_chisq 87
squared_chord 88
squared_euclidean 90
squares 92
topsoe 94
unweighted_unifrac 95
variance_adjusted_unifrac 97
wave_hedges 99
weighted_unifrac 101

Index **103**

`ace` *Abundance-based Coverage Estimator (ACE)*

Description

A non-parametric estimator of species richness that separates features into abundant and rare groups.

Usage

```
ace(counts, cutoff = 10L, margin = 1L, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`cutoff`	The maximum number of observations to consider "rare". Default: `10`.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The ACE metric separates features into "abundant" and "rare" groups based on a cutoff (usually 10 counts). It assumes that the presence of abundant species is certain, while the true number of rare species must be estimated.

Equations:

$$C_{ace} = 1 - \frac{F_1}{X_{rare}}$$

$$\gamma_{ace}^2 = \max\left[\frac{F_{rare}\sum_{i=1}^{r} i(i-1)F_i}{C_{ace}X_{rare}(X_{rare}-1)} - 1, 0\right]$$

$$D_{ace} = F_{abund} + \frac{F_{rare}}{C_{ace}} + \frac{F_1}{C_{ace}}\gamma_{ace}^2$$

Where:

- r : Rare cutoff (default 10). Features with $\leq r$ counts are considered rare.
- F_i : Number of features with exactly i counts.
- F_1 : Number of features where $X_i = 1$ (singletons).
- F_{rare} : Number of rare features where $X_i \leq r$.
- F_{abund} : Number of abundant features where $X_i > r$.
- X_{rare} : Total counts belonging to rare features.
- C_{ace} : The sample abundance coverage estimator.
- γ_{ace}^2 : The estimated coefficient of variation.

Parameter: cutoff The integer threshold distinguishing rare from abundant species. Standard practice is to use 10.

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Chao, A., & Lee, S. M. (1992). Estimating the number of classes via sample coverage. *Journal of the American Statistical Association*, 87(417), 210-217. doi:10.1080/01621459.1992.10475194

See Also

`alpha_div()`, `vignette('adiv')`

Other Richness metrics: `chao1()`, `margalef()`, `menhinick()`, `observed()`, `squares()`

Examples

```
ace(ex_counts)
```

`aitchison` *Aitchison distance*

Description

Calculates the Euclidean distance between centered log-ratio (CLR) transformed abundances.

Usage

```
aitchison(
  counts,
  margin = 1L,
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`pseudocount`	Value added to counts to handle zeros when `norm = 'clr'`. Ignored for other normalization methods. See **Pseudocount** section.
`pairs`	Which combinations of samples should distances be calculated for? The default value (`NULL`) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Aitchison distance is defined as:

$$\sqrt{\sum_{i=1}^{n}[(\ln X_i - X_L) - (\ln Y_i - Y_L)]^2}$$

Where:

- X_i, Y_i : Absolute counts for the i-th feature.
- X_L, Y_L : Mean log of abundances. $X_L = \frac{1}{n}\sum_{i=1}^{n} \ln X_i$.
- n : The number of features.

Base R Equivalent:

```
x <- log((x + pseudocount) / exp(mean(log(x + pseudocount))))
y <- log((y + pseudocount) / exp(mean(log(y + pseudocount))))
sqrt(sum((x-y)^2)) # Euclidean distance
```

Pseudocount

Zeros are undefined in the Aitchison (CLR) transformation. If `pseudocount` is `NULL` (the default) and zeros are detected, the function uses half the minimum non-zero value (`min(x[x>0]) / 2`) and issues a warning.

To suppress the warning, provide an explicit value (e.g., `1`).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Aitchison, J. (1986). The statistical analysis of compositional data. Chapman and Hall. doi:10.1002/bimj.4710300705

Aitchison, J. (1982). The statistical analysis of compositional data. *Journal of the Royal Statistical Society: Series B (Methodological)*, 44(2), 139-160. doi:10.1111/j.25176161.1982.tb01195.x

Costea, P. I., Zeller, G., Sunagawa, S., & Bork, P. (2014). A fair comparison. *Nature Methods*, 11(4), 359. doi:10.1038/nmeth.2897

Gloor, G. B., Macklaim, J. M., Pawlowsky-Glahn, V., & Egozcue, J. J. (2017). Microbiome datasets are compositional: and this is not optional. *Frontiers in Microbiology*, 8, 2224. doi:10.3389/fmicb.2017.02224

Kaul, A., Mandal, S., Davidov, O., & Peddada, S. D. (2017). Analysis of microbiome data in the presence of excess zeros. *Frontiers in Microbiology*, 8, 2114. doi:10.3389/fmicb.2017.02114

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `bhattacharyya()`, `bray()`, `canberra()`, `chebyshev()`, `chord()`, `clark()`, `divergence()`, `euclidean()`, `gower()`, `hellinger()`, `horn()`, `jensen()`, `jsd()`, `lorentzian()`,

manhattan(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
aitchison(ex_counts, pseudocount = 1)
```

---

alpha_div *Alpha Diversity Wrapper Function*

Description

Alpha Diversity Wrapper Function

Usage

```
alpha_div(
  counts,
  metric,
  norm = "percent",
  cutoff = 10L,
  digits = 3L,
  tree = NULL,
  margin = 1L,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
metric	The name of an alpha diversity metric. One of `c('ace', 'berger', 'brillouin', 'chao1', 'faith', 'fisher', 'inv_simpson', 'margalef', 'mcintosh', 'menhinick', 'observed', 'shannon', 'simpson', 'squares')`. Case-insensitive and partial name matching is supported. Programmatic access via `list_metrics('alpha')`.
norm	Normalize the incoming counts. Options are: • `'none'`: No transformation. • `'percent'`: Relative abundance (sample abundances sum to 1). • `'binary'`: Unweighted presence/absence (each count is either 0 or 1). • `'clr'`: Centered log ratio. • `'rclr'`: Robust centered log ratio. Default: `'none'`.
cutoff	The maximum number of observations to consider "rare". Default: `10`.
digits	Precision of the returned values, in number of decimal places. E.g. the default `digits=3` could return `6.392`.

tree	A phylo-class object representing the phylogenetic tree for the OTUs in counts. The OTU identifiers given by colnames(counts) must be present in tree. Can be omitted if a tree is embedded with the counts object or as attr(counts, 'tree').
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

Integer Count Requirements:

A frequent and critical error in alpha diversity analysis is providing the wrong type of data to a metric's formula. Some indices are mathematically defined based on counts of individuals and require raw, integer abundance data. Others are based on proportional abundances and can accept either integer counts (which are then converted to proportions) or pre-normalized proportional data. Using proportional data with a metric that requires integer counts will return an error message.

Requires Integer Counts Only:

- Chao1
- ACE
- Squares Richness Estimator
- Margalef's Index
- Menhinick's Index
- Fisher's Alpha
- Brillouin Index

Can Use Proportional Data:

- Observed Features
- Shannon Index
- Gini-Simpson Index
- Inverse Simpson Index
- Berger-Parker Index
- McIntosh Index
- Faith's PD

Value

A numeric vector.

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom

- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Examples

```
# Example counts matrix
ex_counts

# Shannon diversity values
alpha_div(ex_counts, 'Shannon')

# Chao1 diversity values
alpha_div(ex_counts, 'c')

# Faith PD values
alpha_div(ex_counts, 'faith', tree = ex_tree)
```

---

berger *Berger-Parker Index*

Description

A measure of the numerical importance of the most abundant species.

Usage

```
berger(counts, margin = 1L, cpus = n_cpus())
```

Arguments

counts	A numeric matrix of count data (samples $\times$ features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Berger-Parker index is defined as the proportional abundance of the most dominant feature:

$$\max(P_i)$$

Where:

- P_i : Proportional abundance of the i-th feature.

Base R Equivalent:

```
x <- ex_counts[1,]
p <- x / sum(x)
max(p)
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Berger, W. H., & Parker, F. L. (1970). Diversity of planktonic foraminifera in deep-sea sediments. *Science*, 168(3937), 1345-1347. doi:10.1126/science.168.3937.1345

See Also

`alpha_div()`, `vignette('adiv')`

Other Dominance metrics: `mcintosh()`

Examples

```
berger(ex_counts)
```

`beta_div` *Beta Diversity Wrapper Function*

Description

Beta Diversity Wrapper Function

Usage

```
beta_div(
  counts,
  metric,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  power = 1.5,
  alpha = 0.5,
  tree = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted by some diversity metrics.
metric	The name of a beta diversity metric. One of `c('aitchison', 'bhattacharyya', 'bray', 'canberra', 'chebyshev', 'chord', 'clark', 'divergence', 'euclidean', 'generalized_unifrac', 'gower', 'hamming', 'hellinger', 'horn', 'jaccard', 'jensen', 'jsd', 'lorentzian', 'manhattan', 'matusita', 'minkowski', 'morisita', 'motyka', 'normalized_unifrac', 'ochiai', 'psym_chisq', 'soergel', 'sorensen', 'squared_chisq', 'squared_chord', 'squared_euclidean', 'topsoe', 'unweighted_unifrac', 'variance_adjusted_unifrac', 'wave_hedges', 'weighted_unifrac')`. Flexible matching is supported (see below). Programmatic access via `list_metrics('beta')`.
margin	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
norm	Normalize the incoming counts. Options are: • `'none'`: No transformation. • `'percent'`: Relative abundance (sample abundances sum to 1). • `'binary'`: Unweighted presence/absence (each count is either 0 or 1). • `'clr'`: Centered log ratio. • `'rclr'`: Robust centered log ratio. Default: `'none'`.
pseudocount	Value added to counts to handle zeros when `norm = 'clr'`. Ignored for other normalization methods. See **Pseudocount** section.
power	Only used when `metric = 'minkowski'`. Scaling factor for the magnitude of differences between communities (p). Default: `1.5`
alpha	Only used when `metric = 'generalized_unifrac'`. How much weight to give to relative abundances; a value between 0 and 1, inclusive. Setting `alpha=1` is equivalent to `normalized_unifrac()`.

tree	Only used by phylogeny-aware metrics. A phylo-class object representing the phylogenetic tree for the OTUs in counts. The OTU identifiers given by colnames(counts) must be present in tree. Can be omitted if a tree is embedded with the counts object or as attr(counts, 'tree').
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

List of Beta Diversity Metrics

Option / Function Name	Metric Name
aitchison	Aitchison distance
bhattacharyya	Bhattacharyya distance
bray	Bray-Curtis dissimilarity
canberra	Canberra distance
chebyshev	Chebyshev distance
chord	Chord distance
clark	Clark's divergence distance
divergence	Divergence
euclidean	Euclidean distance
generalized_unifrac	Generalized UniFrac (GUniFrac)
gower	Gower distance
hamming	Hamming distance
hellinger	Hellinger distance
horn	Horn-Morisita dissimilarity
jaccard	Jaccard distance
jensen	Jensen-Shannon distance
jsd	Jesen-Shannon divergence (JSD)
lorentzian	Lorentzian distance
manhattan	Manhattan distance
matusita	Matusita distance
minkowski	Minkowski distance
morisita	Morisita dissimilarity
motyka	Motyka dissimilarity
normalized_unifrac	Normalized Weighted UniFrac
ochiai	Otsuka-Ochiai dissimilarity
psym_chisq	Probabilistic Symmetric Chi-Squared distance
soergel	Soergel distance
sorensen	Dice-Sorensen dissimilarity
squared_chisq	Squared Chi-Squared distance
squared_chord	Squared Chord distance
squared_euclidean	Squared Euclidean distance
topsoe	Topsoe distance
unweighted_unifrac	Unweighted UniFrac

variance_adjusted_unifrac	Variance-Adjusted Weighted UniFrac (VAW-UniFrac)
wave_hedges	Wave Hedges distance
weighted_unifrac	Weighted UniFrac

Flexible name matching

Case insensitive and partial matching. Any runs of non-alpha characters are converted to underscores. E.g. `metric = 'Weighted UniFrac` selects `weighted_unifrac`.

UniFrac names can be shortened to the first letter plus "unifrac". E.g. `uunifrac`, `w_unifrac`, or `V UniFrac`. These also support partial matching.

Finished code should always use the full primary option name to avoid ambiguity with future additions to the metrics list.

Value

A numeric vector.

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The `pseudocount` parameter is only relevant when `norm = 'clr'`.

Zeros are undefined in the centered log-ratio (CLR) transformation. If `norm = 'clr'`, `pseudocount` is `NULL` (the default), and zeros are detected, the function uses half the minimum non-zero value (`min(x[x>0]) / 2`) and issues a warning.

To suppress the warning, provide an explicit value (e.g., `1`).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See `aitchison` for references.

Examples

```
# Example counts matrix
ex_counts

# Bray-Curtis distances
beta_div(ex_counts, 'bray')

# Generalized UniFrac distances
beta_div(ex_counts, 'GUniFrac', tree = ex_tree)
```

`bhattacharyya` *Bhattacharyya distance*

Description

Measures the similarity of two probability distributions.

Usage

```
bhattacharyya(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples $\times$ features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`pairs`	Which combinations of samples should distances be calculated for? The default value (`NULL`) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Bhattacharyya distance is defined as:

$$-\ln \sum_{i=1}^{n} \sqrt{P_i Q_i}$$

Where:

- P_i, Q_i : Proportional abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]; p <- x / sum(x)
y <- ex_counts[2,]; q <- y / sum(y)
-log(sum(sqrt(p * q)))
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Bhattacharyya, A. (1943). On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society*, 35, 99-109.

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bray(), canberra(), chebyshev(), chord(), clark(), divergence(), euclidean(), gower(), hellinger(), horn(), jensen(), jsd(), lorentzian(), manhattan(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
bhattacharyya(ex_counts)
```

---

bray *Bray-Curtis dissimilarity*

Description

A standard ecological metric quantifying the dissimilarity between communities.

Usage

```
bray(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
norm	Normalize the incoming counts. Options are: • 'none': No transformation. • 'percent': Relative abundance (sample abundances sum to 1). • 'binary': Unweighted presence/absence (each count is either 0 or 1). • 'clr': Centered log ratio. • 'rclr': Robust centered log ratio. Default: 'none'.
pseudocount	Value added to counts to handle zeros when norm = 'clr'. Ignored for other normalization methods. See **Pseudocount** section.
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Bray-Curtis dissimilarity is defined as:

$$\frac{\sum_{i=1}^{n} |X_i - Y_i|}{\sum_{i=1}^{n} (X_i + Y_i)}$$

Where:

- X_i, Y_i : Absolute abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
sum(abs(x-y)) / sum(x+y)
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The pseudocount parameter is only relevant when norm = 'clr'.

Zeros are undefined in the centered log-ratio (CLR) transformation. If norm = 'clr', pseudocount is NULL (the default), and zeros are detected, the function uses half the minimum non-zero value (min(x[x>0]) / 2) and issues a warning.

To suppress the warning, provide an explicit value (e.g., 1).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See aitchison for references.

References

Bray, J. R., & Curtis, J. T. (1957). An ordination of the upland forest communities of southern Wisconsin. *Ecological Monographs*, 27(4), 325-349. doi:10.2307/1942268

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bhattacharyya(), canberra(), chebyshev(), chord(), clark(), divergence(), euclidean(), gower(), hellinger(), horn(), jensen(), jsd(), lorentzian(), manhattan(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
bray(ex_counts)
```

`brillouin` *Brillouin Index*

Description

A diversity index derived from information theory, appropriate for fully censused communities.

Usage

```
brillouin(counts, margin = 1L, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples $\times$ features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Brillouin index is defined as:

$$\frac{\ln\left[\left(\sum_{i=1}^{n} X_i\right)!\right] - \sum_{i=1}^{n} \ln\left(X_i!\right)}{\sum_{i=1}^{n} X_i}$$

Where:

- n : The number of features.
- X_i : Integer count of the i-th feature.

Base R Equivalent:

```
x <- ex_counts[1,]
# note: lgamma(x + 1) == log(x!)
(lgamma(sum(x) + 1) - sum(lgamma(x + 1))) / sum(x)
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Brillouin, L. (1956). Science and information theory. Academic Press.

See Also

`alpha_div()`, `vignette('adiv')`

Other Diversity metrics: `fisher()`, `inv_simpson()`, `shannon()`, `simpson()`

Examples

```
brillouin(ex_counts)
```

`canberra` *Canberra distance*

Description

A weighted version of the Manhattan distance, sensitive to differences when both values are small.

Usage

```
canberra(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`norm`	Normalize the incoming counts. Options are: • `'none'`: No transformation. • `'percent'`: Relative abundance (sample abundances sum to 1). • `'binary'`: Unweighted presence/absence (each count is either 0 or 1). • `'clr'`: Centered log ratio. • `'rclr'`: Robust centered log ratio. Default: `'none'`.
`pseudocount`	Value added to counts to handle zeros when `norm = 'clr'`. Ignored for other normalization methods. See **Pseudocount** section.

pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Canberra distance is defined as:

$$\sum_{i=1}^{n} \frac{|X_i - Y_i|}{X_i + Y_i}$$

Where:

- X_i, Y_i : Absolute abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
sum(abs(x-y) / (x+y))
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The pseudocount parameter is only relevant when norm = 'clr'.

Zeros are undefined in the centered log-ratio (CLR) transformation. If norm = 'clr', pseudocount is NULL (the default), and zeros are detected, the function uses half the minimum non-zero value (min(x[x>0]) / 2) and issues a warning.

To suppress the warning, provide an explicit value (e.g., 1).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See aitchison for references.

References

Lance, G. N., & Williams, W. T. (1966). Computer programs for hierarchical polythetic classification ("similarity analyses"). *The Computer Journal*, 9(1), 60-64. doi:10.1093/comjnl/9.1.60

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `aitchison()`, `bhattacharyya()`, `bray()`, `chebyshev()`, `chord()`, `clark()`, `divergence()`, `euclidean()`, `gower()`, `hellinger()`, `horn()`, `jensen()`, `jsd()`, `lorentzian()`, `manhattan()`, `matusita()`, `minkowski()`, `morisita()`, `motyka()`, `psym_chisq()`, `robust_aitchison()`, `soergel()`, `squared_chisq()`, `squared_chord()`, `squared_euclidean()`, `topsoe()`, `wave_hedges()`

Examples

```
canberra(ex_counts)
```

---

chao1 *Chao1 Richness Estimator*

Description

A non-parametric estimator of the lower bound of species richness.

Usage

```
chao1(counts, margin = 1L, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples $\times$ features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Chao1 estimator uses the ratio of singletons to doubletons to estimate the number of missing species:

$$n + \frac{(F_1)^2}{2F_2}$$

Where:

- n : The number of observed features.
- F_1 : Number of features observed once (singletons).

- F_2 : Number of features observed twice (doubletons).

Base R Equivalent:

```
x <- ex_counts[1,]
sum(x>0) + (sum(x == 1) ** 2) / (2 * sum(x == 2))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Chao, A. (1984). Nonparametric estimation of the number of classes in a population. *Scandinavian Journal of Statistics*, 11, 265-270.

See Also

`alpha_div()`, `vignette('adiv')`

Other Richness metrics: `ace()`, `margalef()`, `menhinick()`, `observed()`, `squares()`

Examples

```
chao1(ex_counts)
```

`chebyshev` *Chebyshev distance*

Description

The maximum difference between any single feature across two samples.

Usage

```
chebyshev(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
norm	Normalize the incoming counts. Options are: • 'none': No transformation. • 'percent': Relative abundance (sample abundances sum to 1). • 'binary': Unweighted presence/absence (each count is either 0 or 1). • 'clr': Centered log ratio. • 'rclr': Robust centered log ratio. Default: 'none'.
pseudocount	Value added to counts to handle zeros when norm = 'clr'. Ignored for other normalization methods. See **Pseudocount** section.
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Chebyshev distance is defined as:

$$\max(|X_i - Y_i|)$$

Where:

- X_i, Y_i : Absolute abundances of the i-th feature.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
max(abs(x-y))
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The pseudocount parameter is only relevant when norm = 'clr'.

Zeros are undefined in the centered log-ratio (CLR) transformation. If norm = 'clr', pseudocount is NULL (the default), and zeros are detected, the function uses half the minimum non-zero value (min(x[x>0]) / 2) and issues a warning.

To suppress the warning, provide an explicit value (e.g., 1).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See aitchison for references.

References

Cantrell, C. D. (2000). Modern mathematical methods for physicists and engineers. Cambridge University Press.

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bhattacharyya(), bray(), canberra(), chord(), clark(), divergence(), euclidean(), gower(), hellinger(), horn(), jensen(), jsd(), lorentzian(), manhattan(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
chebyshev(ex_counts)
```

---

chord *Chord distance*

Description

Euclidean distance between normalized vectors.

Usage

```
chord(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Chord distance is defined as:

$$\sqrt{\sum_{i=1}^{n}\left(\frac{X_i}{\sqrt{\sum_{j=1}^{n}X_j^2}}-\frac{Y_i}{\sqrt{\sum_{j=1}^{n}Y_j^2}}\right)^2}$$

Where:

- X_i, Y_i : Absolute counts of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
sqrt(sum(((x / sqrt(sum(x ^ 2))) - (y / sqrt(sum(y ^ 2))))^2))
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Orlóci, L. (1967). An agglomerative method for classification of plant communities. *Journal of Ecology*, 55(1), 193-206. doi:10.2307/2257725

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bhattacharyya(), bray(), canberra(), chebyshev(), clark(), divergence(), euclidean(), gower(), hellinger(), horn(), jensen(), jsd(), lorentzian(), manhattan(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
chord(ex_counts)
```

---

clark *Clark's divergence distance*

Description

Also known as the coefficient of divergence.

Usage

```
clark(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
norm	Normalize the incoming counts. Options are: • 'none': No transformation. • 'percent': Relative abundance (sample abundances sum to 1). • 'binary': Unweighted presence/absence (each count is either 0 or 1). • 'clr': Centered log ratio. • 'rclr': Robust centered log ratio. Default: 'none'.
pseudocount	Value added to counts to handle zeros when norm = 'clr'. Ignored for other normalization methods. See **Pseudocount** section.

pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

Clark's divergence distance is defined as:

$$\sqrt{\sum_{i=1}^{n}\left(\frac{X_i - Y_i}{X_i + Y_i}\right)^2}$$

Where:

- X_i, Y_i : Absolute abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
sqrt(sum((abs(x - y) / (x + y)) ^ 2))
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The pseudocount parameter is only relevant when norm = 'clr'.

Zeros are undefined in the centered log-ratio (CLR) transformation. If norm = 'clr', pseudocount is NULL (the default), and zeros are detected, the function uses half the minimum non-zero value (min(x[x>0]) / 2) and issues a warning.

To suppress the warning, provide an explicit value (e.g., 1).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See aitchison for references.

References

Clark, P. J. (1952). An extension of the coefficient of divergence for use with multiple characters. *Copeia*, 1952(2), 61-64. doi:10.2307/1438598

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `aitchison()`, `bhattacharyya()`, `bray()`, `canberra()`, `chebyshev()`, `chord()`, `divergence()`, `euclidean()`, `gower()`, `hellinger()`, `horn()`, `jensen()`, `jsd()`, `lorentzian()`, `manhattan()`, `matusita()`, `minkowski()`, `morisita()`, `motyka()`, `psym_chisq()`, `robust_aitchison()`, `soergel()`, `squared_chisq()`, `squared_chord()`, `squared_euclidean()`, `topsoe()`, `wave_hedges()`

Examples

```
clark(ex_counts)
```

`divergence` *Divergence*

Description

A probabilistic divergence metric.

Usage

```
divergence(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`norm`	Normalize the incoming counts. Options are: • `'none'`: No transformation. • `'percent'`: Relative abundance (sample abundances sum to 1). • `'binary'`: Unweighted presence/absence (each count is either 0 or 1). • `'clr'`: Centered log ratio.

- `'rclr'`: Robust centered log ratio.

Default: `'none'`.

`pseudocount` Value added to counts to handle zeros when `norm = 'clr'`. Ignored for other normalization methods. See **Pseudocount** section.

`pairs` Which combinations of samples should distances be calculated for? The default value (`NULL`) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.

`cpus` How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

Divergence is defined as:

$$2\sum_{i=1}^{n}\frac{(P_i - Q_i)^2}{(P_i + Q_i)^2}$$

Where:

- P_i, Q_i : Proportional abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]; p <- x / sum(x)
y <- ex_counts[2,]; q <- y / sum(y)
2 * sum((p - q)^2 / (p + q)^2)
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The `pseudocount` parameter is only relevant when `norm = 'clr'`.

Zeros are undefined in the centered log-ratio (CLR) transformation. If `norm = 'clr'`, `pseudocount` is `NULL` (the default), and zeros are detected, the function uses half the minimum non-zero value (`min(x[x>0]) / 2`) and issues a warning.

To suppress the warning, provide an explicit value (e.g., `1`).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See aitchison for references.

References

Cha, S.-H. (2007). Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 1(4), 300–307.

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bhattacharyya(), bray(), canberra(), chebyshev(), chord(), clark(), euclidean(), gower(), hellinger(), horn(), jensen(), jsd(), lorentzian(), manhattan(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
divergence(ex_counts)
```

---

euclidean *Euclidean distance*

Description

The straight-line distance between two points in multidimensional space.

Usage

```
euclidean(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
norm	Normalize the incoming counts. Options are: • 'none': No transformation. • 'percent': Relative abundance (sample abundances sum to 1). • 'binary': Unweighted presence/absence (each count is either 0 or 1). • 'clr': Centered log ratio. • 'rclr': Robust centered log ratio. Default: 'none'.
pseudocount	Value added to counts to handle zeros when norm = 'clr'. Ignored for other normalization methods. See **Pseudocount** section.
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Euclidean distance is defined as:

$$\sqrt{\sum_{i=1}^{n}(X_i - Y_i)^2}$$

Where:

- X_i, Y_i : Absolute abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
sqrt(sum((x-y)^2))
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom

- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The pseudocount parameter is only relevant when norm = 'clr'.

Zeros are undefined in the centered log-ratio (CLR) transformation. If norm = 'clr', pseudocount is NULL (the default), and zeros are detected, the function uses half the minimum non-zero value (min(x[x>0]) / 2) and issues a warning.

To suppress the warning, provide an explicit value (e.g., 1).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See aitchison for references.

References

Legendre, P., & Legendre, L. (2012). Numerical ecology. Elsevier.

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bhattacharyya(), bray(), canberra(), chebyshev(), chord(), clark(), divergence(), gower(), hellinger(), horn(), jensen(), jsd(), lorentzian(), manhattan(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
euclidean(ex_counts)
```

---

ex_counts *Example counts matrix*

Description

Genera found on four human body sites.

Usage

```
ex_counts
```

Format

A matrix of 4 samples (columns) x 6 genera (rows).

Source

Derived from The Human Microbiome Project dataset.

`ex_tree` *Example phylogenetic tree*

Description

Companion tree for `ex_counts`.

Usage

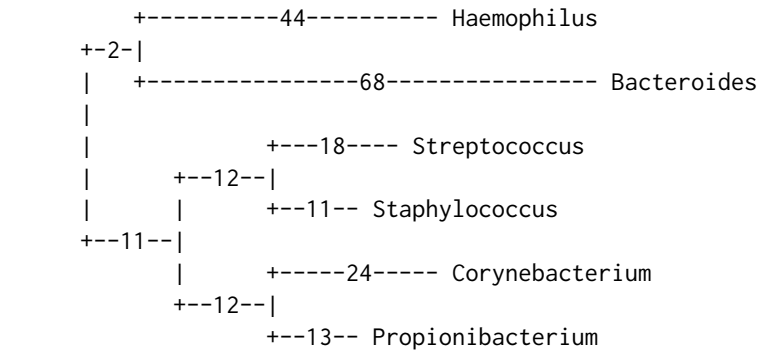
```
ex_tree
```

Format

A `phylo` object.

Details

`ex_tree` encodes this tree structure:



`faith` *Faith's Phylogenetic Diversity (PD)*

Description

Calculates the sum of the branch lengths for all species present in a sample.

Usage

```
faith(counts, tree = NULL, margin = 1L, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples $\times$ features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`tree`	A `phylo`-class object representing the phylogenetic tree for the OTUs in `counts`. The OTU identifiers given by `colnames(counts)` must be present in `tree`. Can be omitted if a tree is embedded with the `counts` object or as `attr(counts, 'tree')`.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

Faith's PD is defined as:

$$\sum_{i=1}^{n} L_i A_i$$

Where:

- n : The number of branches in the phylogenetic tree.
- L_i : The length of the i-th branch.
- A_i : A binary value (1 if any descendants of branch i are present in the sample, 0 otherwise).

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Faith, D. P. (1992). Conservation evaluation and phylogenetic diversity. *Biological Conservation*, 61(1), 1-10. doi:10.1016/00063207(92)912013

See Also

`alpha_div()`, `vignette('adiv')`

Other Phylogenetic metrics: `generalized_unifrac()`, `normalized_unifrac()`, `unweighted_unifrac()`, `variance_adjusted_unifrac()`, `weighted_unifrac()`

Examples

```
faith(ex_counts, tree = ex_tree)
```

---

fisher *Fisher's Alpha*

Description

A parametric diversity index assuming species abundances follow a log-series distribution.

Usage

```
fisher(counts, digits = 3L, margin = 1L, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples $\times$ features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`digits`	Precision of the returned values, in number of decimal places. E.g. the default `digits=3` could return `6.392`.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

Fisher's Alpha (α) is the parameter in the equation:

$$\frac{n}{\alpha} = \ln\left(1 + \frac{X_T}{\alpha}\right)$$

Where:

- n : The number of features.

- X_T : Total of all counts (sequencing depth).

The value of α is solved for iteratively.

Parameter: digits

The precision (number of decimal places) to use when solving the equation.

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Fisher, R. A., Corbet, A. S., & Williams, C. B. (1943). The relation between the number of species and the number of individuals in a random sample of an animal population. *Journal of Animal Ecology*, 12, 42-58. doi:10.2307/1411

See Also

`alpha_div()`, `vignette('adiv')`

Other Diversity metrics: `brillouin()`, `inv_simpson()`, `shannon()`, `simpson()`

Examples

```
fisher(ex_counts)
```

---

generalized_unifrac *Generalized UniFrac (GUniFrac)*

Description

A unified UniFrac distance that balances the weight of abundant and rare lineages.

Usage

```
generalized_unifrac(
  counts,
  tree = NULL,
  alpha = 0.5,
  margin = 1L,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
tree	A phylo-class object representing the phylogenetic tree for the OTUs in counts. The OTU identifiers given by colnames(counts) must be present in tree. Can be omitted if a tree is embedded with the counts object or as attr(counts, 'tree').
alpha	How much weight to give to relative abundances; a value between 0 and 1, inclusive. Setting alpha=1 is equivalent to normalized_unifrac().
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Generalized UniFrac distance is defined as:

$$\frac{\sum_{i=1}^{n} L_i (P_i + Q_i)^{\alpha} \left| \frac{P_i - Q_i}{P_i + Q_i} \right|}{\sum_{i=1}^{n} L_i (P_i + Q_i)^{\alpha}}$$

Where:

- n : The number of branches in the tree.
- L_i : The length of the i-th branch.
- P_i, Q_i : The proportion of the community descending from branch i in sample P and Q.
- α : A scalable weighting factor.

Parameter: alpha

The alpha parameter controls the weight given to abundant lineages. $\alpha = 1$ corresponds to Weighted UniFrac, while $\alpha = 0$ corresponds to Unweighted UniFrac.

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Chen, J., Bittinger, K., Charlson, E. S., Hoffmann, C., Lewis, J., Wu, G. D., ... & Li, H. (2012). Associating microbiome composition with environmental covariates using generalized UniFrac distances. *Bioinformatics*, 28(16), 2106-2113. doi:10.1093/bioinformatics/bts342

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Phylogenetic metrics: `faith()`, `normalized_unifrac()`, `unweighted_unifrac()`, `variance_adjusted_unifrac()`
`weighted_unifrac()`

Examples

```
generalized_unifrac(ex_counts, tree = ex_tree, alpha = 0.5)
```

---

gower *Gower distance*

Description

A distance metric that normalizes differences by the range of the feature.

Usage

```
gower(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
norm	Normalize the incoming counts. Options are: • 'none': No transformation. • 'percent': Relative abundance (sample abundances sum to 1). • 'binary': Unweighted presence/absence (each count is either 0 or 1). • 'clr': Centered log ratio. • 'rclr': Robust centered log ratio. Default: 'none'.
pseudocount	Value added to counts to handle zeros when norm = 'clr'. Ignored for other normalization methods. See **Pseudocount** section.
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Gower distance is defined as:

$$\frac{1}{n}\sum_{i=1}^{n}\frac{|X_i - Y_i|}{R_i}$$

Where:

- X_i, Y_i : Absolute abundances of the i-th feature.
- R_i : The range of the i-th feature across all samples (max - min).
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
r <- abs(x - y)
n <- length(x)
sum(abs(x-y) / r) / n
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq

- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The pseudocount parameter is only relevant when norm = 'clr'.

Zeros are undefined in the centered log-ratio (CLR) transformation. If norm = 'clr', pseudocount is NULL (the default), and zeros are detected, the function uses half the minimum non-zero value (min(x[x>0]) / 2) and issues a warning.

To suppress the warning, provide an explicit value (e.g., 1).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See aitchison for references.

References

Gower, J. C. (1971). A general coefficient of similarity and some of its properties. *Biometrics*, 27(4), 857-871. doi:10.2307/2528823

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bhattacharyya(), bray(), canberra(), chebyshev(), chord(), clark(), divergence(), euclidean(), hellinger(), horn(), jensen(), jsd(), lorentzian(), manhattan(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
gower(ex_counts)
```

---

hamming *Hamming distance*

Description

Measures the minimum number of substitutions required to change one string into the other.

Usage

```
hamming(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Hamming distance is defined as:

$$(A + B) - 2J$$

Where:

- A, B : Number of features in each sample.
- J : Number of features in common (intersection).

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
sum(xor(x, y))
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Hamming, R. W. (1950). Error detecting and error correcting codes. *Bell System Technical Journal*, 29(2), 147-160. doi:10.1002/j.15387305.1950.tb00463.x

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Presence/Absence metrics: jaccard(), ochiai(), sorensen()

Examples

```
hamming(ex_counts)
```

---

hellinger *Hellinger distance*

Description

A distance metric related to the Bhattacharyya distance, often used for community data with many zeros.

Usage

```
hellinger(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Hellinger distance is defined as:

$$\sqrt{\sum_{i=1}^{n}(\sqrt{P_i} - \sqrt{Q_i})^2}$$

Where:

- P_i, Q_i : Proportional abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]; p <- x / sum(x)
y <- ex_counts[2,]; q <- y / sum(y)
sqrt(sum((sqrt(p) - sqrt(q)) ^ 2))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Rao, C. R. (1995). A review of canonical coordinates and an alternative to correspondence analysis using Hellinger distance. *Qüestiió*, 19, 23-63.

Hellinger, E. (1909). Neue Begründung der Theorie quadratischer Formen von unendlichvielen Veränderlichen. *Journal für die reine und angewandte Mathematik*, 136, 210–271. doi:10.1515/crll.1909.136.210

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `aitchison()`, `bhattacharyya()`, `bray()`, `canberra()`, `chebyshev()`, `chord()`, `clark()`, `divergence()`, `euclidean()`, `gower()`, `horn()`, `jensen()`, `jsd()`, `lorentzian()`, `manhattan()`, `matusita()`, `minkowski()`, `morisita()`, `motyka()`, `psym_chisq()`, `robust_aitchison()`, `soergel()`, `squared_chisq()`, `squared_chord()`, `squared_euclidean()`, `topsoe()`, `wave_hedges()`

Examples

```
hellinger(ex_counts)
```

---

horn *Horn-Morisita dissimilarity*

Description

A similarity index based on Simpson's diversity index, suitable for abundance data.

Usage

```
horn(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
norm	Normalize the incoming counts. Options are: • 'none': No transformation. • 'percent': Relative abundance (sample abundances sum to 1). • 'binary': Unweighted presence/absence (each count is either 0 or 1). • 'clr': Centered log ratio. • 'rclr': Robust centered log ratio. Default: 'none'.
pseudocount	Value added to counts to handle zeros when norm = 'clr'. Ignored for other normalization methods. See **Pseudocount** section.
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Horn-Morisita dissimilarity is defined as:

$$1 - \frac{2\sum_{i=1}^{n} P_i Q_i}{\sum_{i=1}^{n} P_i^2 + \sum_{i=1}^{n} Q_i^2}$$

Where:

- P_i, Q_i : Proportional abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
z <- sum(x^2) / sum(x)^2 + sum(y^2) / sum(y)^2
1 - ((2 * sum(x * y)) / (z * sum(x) * sum(y)))
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom

- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The pseudocount parameter is only relevant when norm = 'clr'.

Zeros are undefined in the centered log-ratio (CLR) transformation. If norm = 'clr', pseudocount is NULL (the default), and zeros are detected, the function uses half the minimum non-zero value (min(x[x>0]) / 2) and issues a warning.

To suppress the warning, provide an explicit value (e.g., 1).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See aitchison for references.

References

Horn, H. S. (1966). Measurement of "overlap" in comparative ecological studies. *The American Naturalist*, 100(914), 419-424. doi:10.1086/282436

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bhattacharyya(), bray(), canberra(), chebyshev(), chord(), clark(), divergence(), euclidean(), gower(), hellinger(), jensen(), jsd(), lorentzian(), manhattan(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
horn(ex_counts)
```

---

inv_simpson *Inverse Simpson Index*

Description

A transformation of the Simpson index that represents the "effective number of species".

Usage

```
inv_simpson(counts, margin = 1L, cpus = n_cpus())
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Inverse Simpson index is defined as:

$$1/\sum_{i=1}^{n} P_i^2$$

Where:

- n : The number of features.
- P_i : Proportional abundance of the i-th feature.

Base R Equivalent:

```
x <- ex_counts[1,]
p <- x / sum(x)
1 / sum(p ** 2)
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Simpson, E. H. (1949). Measurement of diversity. *Nature*, 163, 688. doi:10.1038/163688a0

See Also

alpha_div(), vignette('adiv')

Other Diversity metrics: brillouin(), fisher(), shannon(), simpson()

Examples

```
inv_simpson(ex_counts)
```

`jaccard` *Jaccard distance*

Description

Measures dissimilarity between sample sets.

Usage

```
jaccard(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples $\times$ features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`pairs`	Which combinations of samples should distances be calculated for? The default value (`NULL`) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Jaccard distance is defined as:

$$1 - \frac{J}{(A + B - J)}$$

Where:

- A, B : Number of features in each sample.
- J : Number of features in common (intersection).

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
1 - sum(x & y) / sum(x | y)
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Jaccard, P. (1912). The distribution of the flora in the alpine zone. *New Phytologist*, 11(2), 37-50. doi:10.1111/j.14698137.1912.tb05611.x

Jaccard, P. (1908). Nouvelles recherches sur la distribution florale. *Bulletin de la Societe Vaudoise des Sciences Naturelles*, 44(163), 223-270. doi:10.5169/seals268384

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Presence/Absence metrics: `hamming()`, `ochiai()`, `sorensen()`

Examples

```
jaccard(ex_counts)
```

---

jensen *Jensen-Shannon distance*

Description

The square root of the Jensen-Shannon divergence.

Usage

```
jensen(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`pairs`	Which combinations of samples should distances be calculated for? The default value (`NULL`) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Jensen-Shannon distance is defined as:

$$\sqrt{\frac{1}{2}\left[\sum_{i=1}^{n} P_i \ln\left(\frac{2P_i}{P_i+Q_i}\right) + \sum_{i=1}^{n} Q_i \ln\left(\frac{2Q_i}{P_i+Q_i}\right)\right]}$$

Where:

- P_i, Q_i : Proportional abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]; p <- x / sum(x)
y <- ex_counts[2,]; q <- y / sum(y)
sqrt(sum(p * log(2 * p / (p+q)), q * log(2 * q / (p+q))) / 2)
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Endres, D. M., & Schindelin, J. E. (2003). A new metric for probability distributions. *IEEE Transactions on Information Theory*, 49(7), 1858-1860. doi:10.1109/TIT.2003.813506

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bhattacharyya(), bray(), canberra(), chebyshev(), chord(), clark(), divergence(), euclidean(), gower(), hellinger(), horn(), jsd(), lorentzian(), manhattan(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
jensen(ex_counts)
```

---

jsd *Jensen-Shannon divergence (JSD)*

Description

A symmetrized and smoothed version of the Kullback-Leibler divergence.

Usage

```
jsd(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Jensen-Shannon divergence (JSD) is defined as:

$$\frac{1}{2}\left[\sum_{i=1}^{n} P_i \ln\left(\frac{2P_i}{P_i+Q_i}\right) + \sum_{i=1}^{n} Q_i \ln\left(\frac{2Q_i}{P_i+Q_i}\right)\right]$$

Where:

- P_i, Q_i : Proportional abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]; p <- x / sum(x)
y <- ex_counts[2,]; q <- y / sum(y)
sum(p * log(2 * p / (p+q)), q * log(2 * q / (p+q))) / 2
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Lin, J. (1991). Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory*, 37(1), 145-151. doi:10.1109/18.61115

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `aitchison()`, `bhattacharyya()`, `bray()`, `canberra()`, `chebyshev()`, `chord()`, `clark()`, `divergence()`, `euclidean()`, `gower()`, `hellinger()`, `horn()`, `jensen()`, `lorentzian()`, `manhattan()`, `matusita()`, `minkowski()`, `morisita()`, `motyka()`, `psym_chisq()`, `robust_aitchison()`, `soergel()`, `squared_chisq()`, `squared_chord()`, `squared_euclidean()`, `topsoe()`, `wave_hedges()`

Examples

```
jsd(ex_counts)
```

---

list_metrics *Find and Browse Available Metrics*

Description

Programmatic access to the lists of available metrics, and their associated functions.

Usage

```
list_metrics(
  div = c(NA, "alpha", "beta"),
  val = c("data.frame", "list", "func", "id", "name", "div", "phylo", "weighted",
    "int_only", "true_metric"),
  nm = c(NA, "id", "name"),
  phylo = NULL,
  weighted = NULL,
  int_only = NULL,
  true_metric = NULL
)

match_metric(
  metric,
  div = NULL,
  phylo = NULL,
  weighted = NULL,
  int_only = NULL,
  true_metric = NULL
)
```

Arguments

div	Filter by diversity type. One of "alpha" or "beta". Default: NA (no filtering).
val	Sets the return value for this function call. See "Value" section below. Default: "data.frame"
nm	What value to use for the names of the returned object. Default is "id" when val is "list" or "func", otherwise the default is NA (no name).
phylo	Filter by whether a phylogenetic tree is required. TRUE returns only phylogenetic metrics. FALSE returns only non-phylogenetic metrics. Default: NULL (no filtering).
weighted	Filter by whether relative abundance is used. TRUE returns quantitative metrics. FALSE returns qualitative (presence/absence) metrics. Default: NULL (no filtering).
int_only	Filter by whether integer counts are required. TRUE returns metrics requiring integers (e.g. richness estimators). FALSE returns metrics that accept proportions. Default: NULL (no filtering).
true_metric	Filter by whether the metric satisfies the triangle inequality. TRUE returns proper distance metrics. FALSE returns dissimilarities. Default: NULL (no filtering).
metric	The name of an alpha/beta diversity metric to search for. Supports partial matching. All non-alpha characters are ignored.

Value

match_metric()

A list with the following elements.

- name : Metric name, e.g. "Faith's Phylogenetic Diversity"
- id : Metric ID - also the name of the function, e.g. "faith"
- div : Either "alpha" or "beta".
- phylo : TRUE if metric requires a phylogenetic tree; FALSE otherwise.
- weighted : TRUE if metric takes relative abundance into account; FALSE if it only uses presence/absence.
- int_only : TRUE if metric requires integer counts; FALSE otherwise.
- true_metric : TRUE if metric is a true metric and satisfies the triangle inequality; FALSE if it is a non-metric dissimilarity; NA for alpha diversity metrics.
- func : The function for this metric, e.g. ecodive::faith
- params : Formal args for func, e.g. c("counts", "norm", "tree", "cpus")

list_metrics()

The returned object's type and values are controlled with the val and nm arguments.

- val = "data.frame" : The data.frame from which the below options are sourced.
- val = "list" : A list of objects as returned by match_metric() (above).
- val = "func" : A list of functions.
- val = "id" : A character vector of metric IDs.
- val = "name" : A character vector of metric names.
- val = "div" : A character vector "alpha" and/or "beta".
- val = "phylo" : A logical vector indicating which metrics require a tree.
- val = "weighted" : A logical vector indicating which metrics take relative abundance into account (as opposed to just presence/absence).
- val = "int_only" : A logical vector indicating which metrics require integer counts.
- val = "true_metric" : A logical vector indicating which metrics are true metrics and satisfy the triangle inequality, which work better for ordinations such as PCoA.

If nm is set, then the names of the vector or list will be the metric ID (nm="id") or name (nm="name"). When val="data.frame", the names will be applied to the rownames() property of the data.table.

Examples

```
# A data.frame of all available metrics.
head(list_metrics())

# All alpha diversity function names.
list_metrics('alpha', val = 'id')

# Try to find a metric named 'otus'.
m <- match_metric('otus')

# The result is a list that includes the function.
str(m)
```

`lorentzian` *Lorentzian distance*

Description

A log-based distance metric that is robust to outliers.

Usage

```
lorentzian(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`norm`	Normalize the incoming counts. Options are: • `'none'`: No transformation. • `'percent'`: Relative abundance (sample abundances sum to 1). • `'binary'`: Unweighted presence/absence (each count is either 0 or 1). • `'clr'`: Centered log ratio. • `'rclr'`: Robust centered log ratio. Default: `'none'`.
`pseudocount`	Value added to counts to handle zeros when `norm = 'clr'`. Ignored for other normalization methods. See **Pseudocount** section.
`pairs`	Which combinations of samples should distances be calculated for? The default value (`NULL`) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Lorentzian distance is defined as:

$$\sum_{i=1}^{n} \ln\left(1 + |X_i - Y_i|\right)$$

Where:

- X_i, Y_i : Absolute abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
sum(log(1 + abs(x - y)))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The `pseudocount` parameter is only relevant when `norm = 'clr'`.

Zeros are undefined in the centered log-ratio (CLR) transformation. If `norm = 'clr'`, `pseudocount` is `NULL` (the default), and zeros are detected, the function uses half the minimum non-zero value (`min(x[x>0]) / 2`) and issues a warning.

To suppress the warning, provide an explicit value (e.g., `1`).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See `aitchison` for references.

References

Cha, S.-H. (2007). Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 1(4), 300–307.

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `aitchison()`, `bhattacharyya()`, `bray()`, `canberra()`, `chebyshev()`, `chord()`, `clark()`, `divergence()`, `euclidean()`, `gower()`, `hellinger()`, `horn()`, `jensen()`, `jsd()`, `manhattan()`, `matusita()`, `minkowski()`, `morisita()`, `motyka()`, `psym_chisq()`, `robust_aitchison()`, `soergel()`, `squared_chisq()`, `squared_chord()`, `squared_euclidean()`, `topsoe()`, `wave_hedges()`

Examples

```
lorentzian(ex_counts)
```

`manhattan` *Manhattan distance*

Description

The sum of absolute differences, also known as the taxicab geometry.

Usage

```
manhattan(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`norm`	Normalize the incoming counts. Options are: • `'none'`: No transformation. • `'percent'`: Relative abundance (sample abundances sum to 1). • `'binary'`: Unweighted presence/absence (each count is either 0 or 1). • `'clr'`: Centered log ratio. • `'rclr'`: Robust centered log ratio. Default: `'none'`.
`pseudocount`	Value added to counts to handle zeros when `norm = 'clr'`. Ignored for other normalization methods. See **Pseudocount** section.
`pairs`	Which combinations of samples should distances be calculated for? The default value (`NULL`) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Manhattan distance is defined as:

$$\sum_{i=1}^{n} |X_i - Y_i|$$

Where:

- X_i, Y_i : Absolute abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
sum(abs(x-y))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The `pseudocount` parameter is only relevant when `norm = 'clr'`.

Zeros are undefined in the centered log-ratio (CLR) transformation. If `norm = 'clr'`, `pseudocount` is `NULL` (the default), and zeros are detected, the function uses half the minimum non-zero value (`min(x[x>0]) / 2`) and issues a warning.

To suppress the warning, provide an explicit value (e.g., `1`).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See `aitchison` for references.

References

Krause, E. F. (1987). Taxicab geometry: An adventure in non-Euclidean geometry. Dover Publications.

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bhattacharyya(), bray(), canberra(), chebyshev(), chord(), clark(), divergence(), euclidean(), gower(), hellinger(), horn(), jensen(), jsd(), lorentzian(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
manhattan(ex_counts)
```

---

margalef *Margalef's Richness Index*

Description

A richness metric that normalizes the number of species by the log of the total sample size.

Usage

```
margalef(counts, margin = 1L, cpus = n_cpus())
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

Margalef's index is defined as:

$$\frac{n-1}{\ln X_T}$$

Where:

- n : The number of features.
- X_T : Total of all counts (sequencing depth).

Base R Equivalent:

```
x <- ex_counts[1,]
(sum(x > 0) - 1) / log(sum(x))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Margalef, R. (1958). Information theory in ecology. *General Systems*, 3, 36-71.

Gamito, S. (2010). Caution is needed when applying Margalef diversity index. *Ecological Indicators*, 10(2), 550-551. doi:10.1016/j.ecolind.2009.07.006

See Also

`alpha_div()`, `vignette('adiv')`

Other Richness metrics: `ace()`, `chao1()`, `menhinick()`, `observed()`, `squares()`

Examples

```
margalef(ex_counts)
```

`matusita` *Matusita distance*

Description

A distance measure closely related to the Hellinger distance.

Usage

```
matusita(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`pairs`	Which combinations of samples should distances be calculated for? The default value (`NULL`) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Matusita distance is defined as:

$$\sqrt{\sum_{i=1}^{n}\left(\sqrt{P_i}-\sqrt{Q_i}\right)^2}$$

Where:

- P_i, Q_i : Proportional abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]; p <- x / sum(x)
y <- ex_counts[2,]; q <- y / sum(y)
sqrt(sum((sqrt(p) - sqrt(q)) ^ 2))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Matusita, K. (1955). Decision rules, based on the distance, for problems of fit, two samples, and estimation. *The Annals of Mathematical Statistics*, 26(4), 631-640. doi:10.1214/aoms/1177728422

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `aitchison()`, `bhattacharyya()`, `bray()`, `canberra()`, `chebyshev()`, `chord()`, `clark()`, `divergence()`, `euclidean()`, `gower()`, `hellinger()`, `horn()`, `jensen()`, `jsd()`, `lorentzian()`, `manhattan()`, `minkowski()`, `morisita()`, `motyka()`, `psym_chisq()`, `robust_aitchison()`, `soergel()`, `squared_chisq()`, `squared_chord()`, `squared_euclidean()`, `topsoe()`, `wave_hedges()`

Examples

```
matusita(ex_counts)
```

`mcintosh` *McIntosh Index*

Description

A dominance index based on the Euclidean distance from the origin.

Usage

```
mcintosh(counts, margin = 1L, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples $\times$ features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The McIntosh index is defined as:

$$\frac{X_T - \sqrt{\sum_{i=1}^{n}(X_i)^2}}{X_T - \sqrt{X_T}}$$

Where:

- n : The number of features.
- X_i : Integer count of the i-th feature.
- X_T : Total of all counts.

Base R Equivalent:

```
x <- ex_counts[1,]
(sum(x) - sqrt(sum(x^2))) / (sum(x) - sqrt(sum(x)))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

McIntosh, R. P. (1967). An index of diversity and the relation of certain concepts to diversity. *Ecology*, 48(3), 392-404. doi:10.2307/1932674

See Also

`alpha_div()`, `vignette('adiv')`

Other Dominance metrics: `berger()`

Examples

```
mcintosh(ex_counts)
```

`menhinick` *Menhinick's Richness Index*

Description

A richness metric that normalizes the number of species by the square root of the total sample size.

Usage

```
menhinick(counts, margin = 1L, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples $\times$ features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

Menhinick's index is defined as:

$$\frac{n}{\sqrt{X_T}}$$

Where:

- n : The number of features.
- X_T : Total of all counts.

Base R Equivalent:

```
x <- ex_counts[1,]
sum(x > 0) / sqrt(sum(x))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Menhinick, E. F. (1964). A comparison of some species-individuals diversity indices applied to samples of field insects. *Ecology*, 45(4), 859-861. doi:10.2307/1934933

See Also

`alpha_div()`, `vignette('adiv')`

Other Richness metrics: `ace()`, `chao1()`, `margalef()`, `observed()`, `squares()`

Examples

```
menhinick(ex_counts)
```

`minkowski` *Minkowski distance*

Description

A generalized metric that includes Euclidean and Manhattan distance as special cases.

Usage

```
minkowski(
  counts,
  margin = 1L,
  power = 1.5,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`power`	Scaling factor for the magnitude of differences between communities (p). Default: `1.5`
`norm`	Normalize the incoming counts. Options are: • `'none'`: No transformation. • `'percent'`: Relative abundance (sample abundances sum to 1). • `'binary'`: Unweighted presence/absence (each count is either 0 or 1). • `'clr'`: Centered log ratio. • `'rclr'`: Robust centered log ratio. Default: `'none'`.
`pseudocount`	Value added to counts to handle zeros when `norm = 'clr'`. Ignored for other normalization methods. See **Pseudocount** section.
`pairs`	Which combinations of samples should distances be calculated for? The default value (`NULL`) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Minkowski distance is defined as:

$$\sqrt[p]{\sum_{i=1}^{n}(X_i - Y_i)^p}$$

Where:

- X_i, Y_i : Absolute abundances of the i-th feature.
- n : The number of features.
- p : The geometry of the space (power parameter).

Parameter: power

The `power` parameter (default 1.5) determines the value of p in the equation.

Special Cases

- **Manhattan distance**: When $p = 1$, the formula reduces to the sum of absolute differences.
- **Euclidean distance**: When $p = 2$, the formula reduces to the standard straight-line distance.
- **Chebyshev distance**: When $p \to \infty$, the formula reduces to the maximum absolute difference.

Base R Equivalent:

```
p <- 1.5
x <- ex_counts[1,]
y <- ex_counts[2,]
sum(abs(x - y)^p) ^ (1/p)
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The `pseudocount` parameter is only relevant when `norm = 'clr'`.

Zeros are undefined in the centered log-ratio (CLR) transformation. If `norm = 'clr'`, `pseudocount` is `NULL` (the default), and zeros are detected, the function uses half the minimum non-zero value (`min(x[x>0]) / 2`) and issues a warning.

To suppress the warning, provide an explicit value (e.g., `1`).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See `aitchison` for references.

References

Deza, M. M., & Deza, E. (2009). Encyclopedia of distances. Springer.

Minkowski, H. (1896). *Geometrie der Zahlen*. Teubner.

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `aitchison()`, `bhattacharyya()`, `bray()`, `canberra()`, `chebyshev()`, `chord()`, `clark()`, `divergence()`, `euclidean()`, `gower()`, `hellinger()`, `horn()`, `jensen()`, `jsd()`, `lorentzian()`, `manhattan()`, `matusita()`, `morisita()`, `motyka()`, `psym_chisq()`, `robust_aitchison()`, `soergel()`, `squared_chisq()`, `squared_chord()`, `squared_euclidean()`, `topsoe()`, `wave_hedges()`

Examples

```
minkowski(ex_counts, power = 2) # Equivalent to Euclidean
```

`morisita` *Morisita dissimilarity*

Description

A measure of overlap between samples that is independent of sample size. Requires integer counts.

Usage

```
morisita(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`pairs`	Which combinations of samples should distances be calculated for? The default value (`NULL`) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Morisita dissimilarity is defined as:

$$1 - \frac{2\sum_{i=1}^{n} X_i Y_i}{\left(\frac{\sum_{i=1}^{n} X_i(X_i-1)}{X_T(X_T-1)} + \frac{\sum_{i=1}^{n} Y_i(Y_i-1)}{Y_T(Y_T-1)}\right) X_T Y_T}$$

Where:

- X_i, Y_i : Absolute counts of the i-th feature.
- X_T, Y_T : Total counts in each sample. $X_T = \sum_{i=1}^{n} X_i$.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
simpson_x <- sum(x * (x - 1)) / (sum(x) * (sum(x) - 1))
simpson_y <- sum(y * (y - 1)) / (sum(y) * (sum(y) - 1))
1 - ((2 * sum(x * y)) / ((simpson_x + simpson_y) * sum(x) * sum(y)))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Morisita, M. (1959). Measuring of interspecific association and similarity between communities. *Memoirs of the Faculty of Science, Kyushu University, Series E (Biology)*, 3, 65-80.

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `aitchison()`, `bhattacharyya()`, `bray()`, `canberra()`, `chebyshev()`, `chord()`, `clark()`, `divergence()`, `euclidean()`, `gower()`, `hellinger()`, `horn()`, `jensen()`, `jsd()`, `lorentzian()`, `manhattan()`, `matusita()`, `minkowski()`, `motyka()`, `psym_chisq()`, `robust_aitchison()`, `soergel()`, `squared_chisq()`, `squared_chord()`, `squared_euclidean()`, `topsoe()`, `wave_hedges()`

Examples

```
morisita(ex_counts)
```

`motyka` *Motyka dissimilarity*

Description

Also known as the Bray-Curtis dissimilarity when applied to abundance data, but formulated slightly differently.

Usage

```
motyka(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
norm	Normalize the incoming counts. Options are: • 'none': No transformation. • 'percent': Relative abundance (sample abundances sum to 1). • 'binary': Unweighted presence/absence (each count is either 0 or 1). • 'clr': Centered log ratio. • 'rclr': Robust centered log ratio. Default: 'none'.
pseudocount	Value added to counts to handle zeros when norm = 'clr'. Ignored for other normalization methods. See **Pseudocount** section.
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Motyka dissimilarity is defined as:

$$\frac{\sum_{i=1}^{n} \max(X_i, Y_i)}{\sum_{i=1}^{n} (X_i + Y_i)}$$

Where:

- X_i, Y_i : Absolute abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
sum(pmax(x, y)) / sum(x, y)
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment

- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The pseudocount parameter is only relevant when norm = 'clr'.

Zeros are undefined in the centered log-ratio (CLR) transformation. If norm = 'clr', pseudocount is NULL (the default), and zeros are detected, the function uses half the minimum non-zero value (min(x[x>0]) / 2) and issues a warning.

To suppress the warning, provide an explicit value (e.g., 1).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See aitchison for references.

References

Motyka, J. (1947). O celach i metodach badan geobotanicznych. *Annales Universitatis Mariae Curie-Sklodowska, Sectio C*, 3, 1-168.

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bhattacharyya(), bray(), canberra(), chebyshev(), chord(), clark(), divergence(), euclidean(), gower(), hellinger(), horn(), jensen(), jsd(), lorentzian(), manhattan(), matusita(), minkowski(), morisita(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
motyka(ex_counts)
```

---

n_cpus *Number of CPU Cores*

Description

A thin wrapper around parallely::availableCores(). If the parallely package is not installed, then it falls back to parallel::detectCores(all.tests = TRUE, logical = TRUE). Returns 1 if pthread support is unavailable or when the number of cpus cannot be determined.

Usage

```
n_cpus()
```

Value

A scalar integer, guaranteed to be at least 1.

Examples

```
n_cpus()
```

---

normalized_unifrac *Normalized Weighted UniFrac*

Description

Weighted UniFrac normalized by the tree length to allow comparison between trees.

Usage

```
normalized_unifrac(
  counts,
  tree = NULL,
  margin = 1L,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
tree	A phylo-class object representing the phylogenetic tree for the OTUs in counts. The OTU identifiers given by colnames(counts) must be present in tree. Can be omitted if a tree is embedded with the counts object or as attr(counts, 'tree').
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Normalized Weighted UniFrac distance is defined as:

$$\frac{\sum_{i=1}^{n} L_i |P_i - Q_i|}{\sum_{i=1}^{n} L_i (P_i + Q_i)}$$

Where:

- n : The number of branches in the tree.
- L_i : The length of the i-th branch.
- P_i, Q_i : The proportion of the community descending from branch i in sample P and Q.

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Lozupone, C. A., Hamady, M., Kelley, S. T., & Knight, R. (2007). Quantitative and qualitative beta diversity measures lead to different insights into factors that structure microbial communities. *Applied and Environmental Microbiology*, 73(5), 1576-1585. doi:10.1128/AEM.0199606

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Phylogenetic metrics: `faith()`, `generalized_unifrac()`, `unweighted_unifrac()`, `variance_adjusted_unifrac()`, `weighted_unifrac()`

Examples

```
normalized_unifrac(ex_counts, tree = ex_tree)
```

`observed` *Observed Features*

Description

The count of unique features (richness) in a sample.

Usage

```
observed(counts, margin = 1L, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

Observed features is defined simply as the number of features with non-zero abundance:

$$n$$

Base R Equivalent:

```
x <- ex_counts[1,]
sum(x > 0)
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

See Also

`alpha_div()`, `vignette('adiv')`

Other Richness metrics: `ace()`, `chao1()`, `margalef()`, `menhinick()`, `squares()`

Examples

```
observed(ex_counts)
```

`ochiai` *Otsuka-Ochiai dissimilarity*

Description

Also known as the cosine similarity for binary data.

Usage

```
ochiai(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`pairs`	Which combinations of samples should distances be calculated for? The default value (`NULL`) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Otsuka-Ochiai dissimilarity is defined as:

$$1 - \frac{J}{\sqrt{AB}}$$

Where:

- A, B : Number of features in each sample.
- J : Number of features in common (intersection).

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
1 - sum(x & y) / sqrt(sum(x>0) * sum(y>0))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Ochiai, A. (1957). Zoogeographic studies on the soleoid fishes found in Japan and its neighbouring regions. *Bulletin of the Japanese Society of Scientific Fisheries*, 22, 526-530.

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Presence/Absence metrics: `hamming()`, `jaccard()`, `sorensen()`

Examples

```
ochiai(ex_counts)
```

---

psym_chisq *Probabilistic Symmetric Chi-Squared distance*

Description

A chi-squared based distance metric for comparing probability distributions.

Usage

```
psym_chisq(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`pairs`	Which combinations of samples should distances be calculated for? The default value (`NULL`) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Probabilistic Symmetric χ^2 distance is defined as:

$$2\sum_{i=1}^{n}\frac{(P_i - Q_i)^2}{P_i + Q_i}$$

Where:

- P_i, Q_i : Proportional abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]; p <- x / sum(x)
y <- ex_counts[2,]; q <- y / sum(y)
2 * sum((p - q)^2 / (p + q))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Cha, S.-H. (2007). Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 1(4), 300–307.

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `aitchison()`, `bhattacharyya()`, `bray()`, `canberra()`, `chebyshev()`, `chord()`, `clark()`, `divergence()`, `euclidean()`, `gower()`, `hellinger()`, `horn()`, `jensen()`, `jsd()`, `lorentzian()`, `manhattan()`, `matusita()`, `minkowski()`, `morisita()`, `motyka()`, `robust_aitchison()`, `soergel()`, `squared_chisq()`, `squared_chord()`, `squared_euclidean()`, `topsoe()`, `wave_hedges()`

Examples

```
psym_chisq(ex_counts)
```

`rarefy` *Rarefy Observation Counts*

Description

Sub-sample observations from a feature table such that all samples have the same library size (depth). This is performed via random sampling without replacement.

Usage

```
rarefy(
  counts,
  depth = NULL,
  seed = 0,
  times = NULL,
  drop = TRUE,
  margin = 1L,
  cpus = n_cpus(),
  warn = interactive()
)
```

Arguments

`counts`	A numeric matrix or sparse matrix object (e.g., `dgCMatrix`). Counts must be integers.
`depth`	The number of observations to keep per sample. If `NULL` (the default), a depth is auto-selected to maximize data retention.
`seed`	An integer seed for the random number generator. Providing the same seed guarantees reproducible results. Default: `0`
`times`	The number of independent rarefactions to perform. If set, returns a list of matrices. Seeds for subsequent iterations are sequential (`seed`, `seed + 1`, ...). Default: `NULL`
`drop`	Logical. If `TRUE`, samples with fewer than `depth` observations are discarded. If `FALSE`, they are kept with their original counts. Default: `TRUE`
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.
`warn`	Logical. If `TRUE`, emits a warning when samples are dropped or returned unrarefied due to insufficient depth. Default: `interactive()`

Value

A rarefied matrix. The output class (`matrix`, `dgCMatrix`, etc.) matches the input class.

Auto-Depth Selection

If `depth` is `NULL`, the function defaults to the highest depth that retains at least 10% of the total observations in the dataset.

Dropping vs. Retaining Samples

If a sample has fewer observations than the specified `depth`:

- `drop = TRUE` (Default): The sample is removed from the output matrix.
- `drop = FALSE`: The sample is returned **unmodified** (with its original counts). It is *not* rarefied or zeroed out.

Zero-Sum Features

Features (OTUs, ASVs, Genes) that lose all observations during rarefaction are **always retained** as columns/rows of zeros. This ensures the output matrix dimensions remain consistent with the input (barring dropped samples).

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Examples

```
# A 4-sample x 5-OTU matrix with samples in rows.
counts <- matrix(c(0,0,0,0,0,8,9,10,5,5,5,5,2,0,0,0,6,5,7,0), 4, 5,
  dimnames = list(LETTERS[1:4], paste0('OTU', 1:5)))
counts
rowSums(counts)

# Rarefy all samples to a depth of 18.
# Sample 'A' (13 counts) and 'D' (15 counts) will be dropped.
r_mtx <- rarefy(counts, depth = 18)
r_mtx
rowSums(r_mtx)

# Keep under-sampled samples by setting `drop = FALSE`.
# Samples 'A' and 'D' are returned with their original counts.
r_mtx <- rarefy(counts, depth = 18, drop = FALSE)
r_mtx
rowSums(r_mtx)
```

```
# Perform 3 independent rarefactions.
r_list <- rarefy(counts, times = 3)
length(r_list)

# Sparse matrices are supported and their class is preserved.
if (requireNamespace('Matrix', quietly = TRUE)) {
  counts_dgC <- Matrix::Matrix(counts, sparse = TRUE)
  str(rarefy(counts_dgC))
}
```

`read_tree` *Read a newick formatted phylogenetic tree.*

Description

A phylogenetic tree is required for computing UniFrac distance matrices. You can load a tree from a file or by providing the tree string directly. This tree must be in Newick format, also known as parenthetic format and New Hampshire format.

Usage

```
read_tree(newick, underscores = FALSE)
```

Arguments

`newick`	Input data as either a file path, URL, or Newick string. Compressed (gzip or bzip2) files are also supported.
`underscores`	If `TRUE`, underscores in unquoted names will remain underscores. If `FALSE`, underscores in unquoted named will be converted to spaces.

Value

A `phylo` class object representing the tree.

Examples

```
tree <- read_tree("
    (A:0.99,((B:0.87,C:0.89):0.51,(((D:0.16,(E:0.83,F:0.96)
    :0.94):0.69,(G:0.92,(H:0.62,I:0.85):0.54):0.23):0.74,J:0.1
    2):0.43):0.67);")
class(tree)
```

`robust_aitchison` *Robust Aitchison distance*

Description

Calculates the pairwise Robust Aitchison distance for compositional data. This method is specifically engineered for sparse datasets—such as microbiome OTU/ASV tables—by calculating distances based only on observed positive abundances, entirely avoiding the need for arbitrary pseudocounts.

Usage

```
robust_aitchison(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`pairs`	Which combinations of samples should distances be calculated for? The default value (`NULL`) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Robust Aitchison distance is defined as:

$$\sqrt{\sum_{i=1}^{n}(X_i^* - Y_i^*)^2}$$

Where:

- X_i^*, Y_i^* : The rclr-transformed counts for the i-th feature. For a given sample X, $X_i^* = \ln(X_i) - X_L$ if $X_i > 0$, and 0 otherwise.
- X_L, Y_L : Mean log of strictly positive abundances. $X_L = \frac{1}{|P_X|}\sum_{j \in P_X} \ln X_j$, where P_X is the set of indices where $X > 0$.
- $|P_X|$, $|P_Y|$: The number of strictly positive features in the respective samples.
- n : The total number of features.

Base R Equivalent:

```
x <- ifelse(x > 0, log(x) - mean(log(x[x > 0])), 0)
y <- ifelse(y > 0, log(y) - mean(log(y[y > 0])), 0)
sqrt(sum((x-y)^2)) # Euclidean distance
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Martino, C., Morton, J. T., Marotz, C. A., Thompson, L. R., Tripathi, A., Knight, R., and Zengler, K. (2019). A novel sparse compositional technique reveals microbial perturbations. *mSystems*, 4(1), e00016-19. doi:10.1128/mSystems.0001619

See Also

`beta_div()`, `aitchison()`

`vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `aitchison()`, `bhattacharyya()`, `bray()`, `canberra()`, `chebyshev()`, `chord()`, `clark()`, `divergence()`, `euclidean()`, `gower()`, `hellinger()`, `horn()`, `jensen()`, `jsd()`, `lorentzian()`, `manhattan()`, `matusita()`, `minkowski()`, `morisita()`, `motyka()`, `psym_chisq()`, `soergel()`, `squared_chisq()`, `squared_chord()`, `squared_euclidean()`, `topsoe()`, `wave_hedges()`

Examples

```
robust_aitchison(ex_counts)
```

---

shannon *Shannon Diversity Index*

Description

A commonly used diversity index accounting for both abundance and evenness.

Usage

```
shannon(counts, margin = 1L, cpus = n_cpus())
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Shannon index (entropy) is defined as:

$$-\sum_{i=1}^{n} P_i \times \ln(P_i)$$

Where:

- n : The number of features.
- P_i : Proportional abundance of the i-th feature.

Base R Equivalent:

```
x <- ex_counts[1,]
p <- x / sum(x)
-sum(p * log(p))
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27, 379-423.

Shannon, C. E., & Weaver, W. (1949). *The Mathematical Theory of Communication*. University of Illinois Press.

See Also

alpha_div(), vignette('adiv')

Other Diversity metrics: brillouin(), fisher(), inv_simpson(), simpson()

Examples

```
shannon(ex_counts)
```

---

simpson *Gini-Simpson Index*

Description

The probability that two entities taken at random from the dataset represent different types.

Usage

```
simpson(counts, margin = 1L, cpus = n_cpus())
```

Arguments

counts	A numeric matrix of count data (samples $\times$ features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Gini-Simpson index is defined as:

$$1 - \sum_{i=1}^{n} P_i^2$$

Where:

- n : The number of features.
- P_i : Proportional abundance of the i-th feature.

Base R Equivalent:

```
x <- ex_counts[1,]
p <- x / sum(x)
1 - sum(p ** 2)
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Simpson, E. H. (1949). Measurement of diversity. *Nature*, 163, 688. doi:10.1038/163688a0

See Also

`alpha_div()`, `vignette('adiv')`

Other Diversity metrics: `brillouin()`, `fisher()`, `inv_simpson()`, `shannon()`

Examples

```
simpson(ex_counts)
```

`soergel` *Soergel distance*

Description

A distance metric related to the Bray-Curtis and Jaccard indices.

Usage

```
soergel(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
norm	Normalize the incoming counts. Options are: • 'none': No transformation. • 'percent': Relative abundance (sample abundances sum to 1). • 'binary': Unweighted presence/absence (each count is either 0 or 1). • 'clr': Centered log ratio. • 'rclr': Robust centered log ratio. Default: 'none'.
pseudocount	Value added to counts to handle zeros when norm = 'clr'. Ignored for other normalization methods. See **Pseudocount** section.
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Soergel distance is defined as:

$$\frac{\sum_{i=1}^{n} |X_i - Y_i|}{\sum_{i=1}^{n} \max(X_i, Y_i)}$$

Where:

- X_i, Y_i : Absolute abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
sum(abs(x - y)) / sum(pmax(x, y))
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment

- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The pseudocount parameter is only relevant when norm = 'clr'.

Zeros are undefined in the centered log-ratio (CLR) transformation. If norm = 'clr', pseudocount is NULL (the default), and zeros are detected, the function uses half the minimum non-zero value (min(x[x>0]) / 2) and issues a warning.

To suppress the warning, provide an explicit value (e.g., 1).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See aitchison for references.

References

Cha, S.-H. (2007). Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 1(4), 300–307.

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bhattacharyya(), bray(), canberra(), chebyshev(), chord(), clark(), divergence(), euclidean(), gower(), hellinger(), horn(), jensen(), jsd(), lorentzian(), manhattan(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
soergel(ex_counts)
```

---

sorensen *Dice-Sorensen dissimilarity*

Description

A statistic used for comparing the similarity of two samples.

Usage

```
sorensen(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Dice-Sorensen dissimilarity is defined as:

$$\frac{2J}{(A+B)}$$

Where:

- A, B : Number of features in each sample.
- J : Number of features in common (intersection).

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
2 * sum(x & y) / sum(x>0, y>0)
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Sørensen, T. (1948). A method of establishing groups of equal amplitude in plant sociology based on similarity of species content. *Kongelige Danske Videnskabernes Selskab, Biologiske Skrifter*, 5, 1-34.

Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, 26(3), 297–302. doi:10.2307/1932409

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Presence/Absence metrics: hamming(), jaccard(), ochiai()

Examples

```
sorensen(ex_counts)
```

---

squared_chisq *Squared Chi-Squared distance*

Description

The squared version of the Chi-Squared distance.

Usage

```
squared_chisq(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Squared χ^2 distance is defined as:

$$\sum_{i=1}^{n} \frac{(P_i - Q_i)^2}{P_i + Q_i}$$

Where:

- P_i, Q_i : Proportional abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]; p <- x / sum(x)
y <- ex_counts[2,]; q <- y / sum(y)
sum((p - q)^2 / (p + q))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Cha, S.-H. (2007). Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 1(4), 300–307.

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `aitchison()`, `bhattacharyya()`, `bray()`, `canberra()`, `chebyshev()`, `chord()`, `clark()`, `divergence()`, `euclidean()`, `gower()`, `hellinger()`, `horn()`, `jensen()`, `jsd()`, `lorentzian()`, `manhattan()`, `matusita()`, `minkowski()`, `morisita()`, `motyka()`, `psym_chisq()`, `robust_aitchison()`, `soergel()`, `squared_chord()`, `squared_euclidean()`, `topsoe()`, `wave_hedges()`

Examples

```
squared_chisq(ex_counts)
```

`squared_chord` *Squared Chord distance*

Description

The squared version of the Chord distance.

Usage

```
squared_chord(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Squared Chord distance is defined as:

$$\sum_{i=1}^{n} \left(\sqrt{P_i} - \sqrt{Q_i} \right)^2$$

Where:

- P_i, Q_i : Proportional abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]; p <- x / sum(x)
y <- ex_counts[2,]; q <- y / sum(y)
sum((sqrt(x) - sqrt(y)) ^ 2)
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Legendre, P., & Legendre, L. (2012). Numerical ecology. Elsevier.

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bhattacharyya(), bray(), canberra(), chebyshev(), chord(), clark(), divergence(), euclidean(), gower(), hellinger(), horn(), jensen(), jsd(), lorentzian(), manhattan(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_euclidean(), topsoe(), wave_hedges()

Examples

```
squared_chord(ex_counts)
```

---

squared_euclidean *Squared Euclidean distance*

Description

The squared Euclidean distance between two vectors.

Usage

```
squared_euclidean(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
norm	Normalize the incoming counts. Options are: • 'none': No transformation. • 'percent': Relative abundance (sample abundances sum to 1). • 'binary': Unweighted presence/absence (each count is either 0 or 1). • 'clr': Centered log ratio. • 'rclr': Robust centered log ratio. Default: 'none'.
pseudocount	Value added to counts to handle zeros when norm = 'clr'. Ignored for other normalization methods. See **Pseudocount** section.

pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Squared Euclidean distance is defined as:

$$\sum_{i=1}^{n}(X_i - Y_i)^2$$

Where:

- X_i, Y_i : Absolute abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
sum((x-y)^2)
```

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The pseudocount parameter is only relevant when norm = 'clr'.

Zeros are undefined in the centered log-ratio (CLR) transformation. If norm = 'clr', pseudocount is NULL (the default), and zeros are detected, the function uses half the minimum non-zero value (min(x[x>0]) / 2) and issues a warning.

To suppress the warning, provide an explicit value (e.g., 1).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See aitchison for references.

References

Legendre, P., & Legendre, L. (2012). Numerical ecology. Elsevier.

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `aitchison()`, `bhattacharyya()`, `bray()`, `canberra()`, `chebyshev()`, `chord()`, `clark()`, `divergence()`, `euclidean()`, `gower()`, `hellinger()`, `horn()`, `jensen()`, `jsd()`, `lorentzian()`, `manhattan()`, `matusita()`, `minkowski()`, `morisita()`, `motyka()`, `psym_chisq()`, `robust_aitchison()`, `soergel()`, `squared_chisq()`, `squared_chord()`, `topsoe()`, `wave_hedges()`

Examples

```
squared_euclidean(ex_counts)
```

---

squares *Squares Richness Estimator*

Description

A richness estimator based on the concept of "squares" (counts of species observed once or twice).

Usage

```
squares(counts, margin = 1L, cpus = n_cpus())
```

Arguments

`counts`	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
`margin`	The margin containing samples. `1` if samples are rows, `2` if samples are columns. Ignored when `counts` is a special object class (e.g. `phyloseq`). Default: `1`
`cpus`	How many parallel processing threads should be used. The default, `n_cpus()`, will use all logical CPU cores.

Details

The Squares estimator is defined as:

$$n + \frac{(F_1)^2 \sum_{i=1}^{n} (X_i)^2}{X_T^2 - nF_1}$$

Where:

- n : The number of observed features.
- X_T : Total of all counts.

- F_1 : Number of features observed once (singletons).
- X_i : Integer count of the i-th feature.

Base R Equivalent:

```
x  <- ex_counts[1,]
N  <- sum(x)      # sampling depth
S  <- sum(x > 0)  # observed features
F1 <- sum(x == 1) # singletons
S + ((sum(x^2) * (F1^2)) / ((N^2) - F1 * S))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Alroy, J. (2018). Limits to species richness estimates based on subsampling. *Paleobiology*, 44(2), 177-194. doi:10.1017/pab.2017.38

See Also

`alpha_div()`, `vignette('adiv')`

Other Richness metrics: `ace()`, `chao1()`, `margalef()`, `menhinick()`, `observed()`

Examples

```
squares(ex_counts)
```

topsoe *Topsoe distance*

Description

A symmetric divergence measure based on the Jensen-Shannon divergence.

Usage

```
topsoe(counts, margin = 1L, pairs = NULL, cpus = n_cpus())
```

Arguments

counts	A numeric matrix of count data (samples $\times$ features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Topsoe distance is defined as:

$$\sum_{i=1}^{n} P_i \ln\left(\frac{2P_i}{P_i + Q_i}\right) + \sum_{i=1}^{n} Q_i \ln\left(\frac{2Q_i}{P_i + Q_i}\right)$$

Where:

- P_i, Q_i : Proportional abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]; p <- x / sum(x)
y <- ex_counts[2,]; q <- y / sum(y)
sum(p * log(2 * p / (p+q)), q * log(2 * y / (p+q)))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Topsoe, F. (2000). Some inequalities for information divergence and related measures of discrimination. *IEEE Transactions on Information Theory*, 46(4), 1602-1609. doi:10.1109/18.850703

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Abundance metrics: `aitchison()`, `bhattacharyya()`, `bray()`, `canberra()`, `chebyshev()`, `chord()`, `clark()`, `divergence()`, `euclidean()`, `gower()`, `hellinger()`, `horn()`, `jensen()`, `jsd()`, `lorentzian()`, `manhattan()`, `matusita()`, `minkowski()`, `morisita()`, `motyka()`, `psym_chisq()`, `robust_aitchison()`, `soergel()`, `squared_chisq()`, `squared_chord()`, `squared_euclidean()`, `wave_hedges()`

Examples

```
topsoe(ex_counts)
```

---

unweighted_unifrac *Unweighted UniFrac*

Description

A phylogenetic distance metric that accounts for the presence/absence of lineages.

Usage

```
unweighted_unifrac(
  counts,
  tree = NULL,
  margin = 1L,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
tree	A phylo-class object representing the phylogenetic tree for the OTUs in counts. The OTU identifiers given by colnames(counts) must be present in tree. Can be omitted if a tree is embedded with the counts object or as attr(counts, 'tree').
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Unweighted UniFrac distance is defined as:

$$\frac{1}{n}\sum_{i=1}^{n} L_i |A_i - B_i|$$

Where:

- n : The number of branches in the tree.
- L_i : The length of the i-th branch.
- A_i, B_i : Binary values (0 or 1) indicating if descendants of branch i are present in sample A or B.

Input Types

The counts parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- phyloseq
- rbiom
- SummarizedExperiment
- TreeSummarizedExperiment

For large datasets, standard matrix operations may be slow. See vignette('performance') for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Lozupone, C., & Knight, R. (2005). UniFrac: a new phylogenetic method for comparing microbial communities. *Applied and Environmental Microbiology*, 71(12), 8228-8235. doi:10.1128/AEM.71.12.82288235.2005

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Phylogenetic metrics: faith(), generalized_unifrac(), normalized_unifrac(), variance_adjusted_unifrac(), weighted_unifrac()

Examples

```
unweighted_unifrac(ex_counts, tree = ex_tree)
```

---

variance_adjusted_unifrac

Variance-Adjusted Weighted UniFrac

Description

A weighted UniFrac that adjusts for the expected variance of the metric.

Usage

```
variance_adjusted_unifrac(
  counts,
  tree = NULL,
  margin = 1L,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
tree	A phylo-class object representing the phylogenetic tree for the OTUs in counts. The OTU identifiers given by colnames(counts) must be present in tree. Can be omitted if a tree is embedded with the counts object or as attr(counts, 'tree').
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Variance-Adjusted Weighted UniFrac distance is defined as:

$$\frac{\sum_{i=1}^{n} L_i \frac{|P_i - Q_i|}{\sqrt{(P_i + Q_i)(2 - P_i - Q_i)}}}{\sum_{i=1}^{n} L_i \frac{P_i + Q_i}{\sqrt{(P_i + Q_i)(2 - P_i - Q_i)}}}$$

Where:

- n : The number of branches in the tree.
- L_i : The length of the i-th branch.
- P_i, Q_i : The proportion of the community descending from branch i in sample P and Q.

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Chang, Q., Luan, Y., & Sun, F. (2011). Variance adjusted weighted UniFrac: a powerful beta diversity measure for comparing communities based on phylogeny. *BMC Bioinformatics*, 12, 118. doi:10.1186/1471210512118

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Phylogenetic metrics: `faith()`, `generalized_unifrac()`, `normalized_unifrac()`, `unweighted_unifrac()`, `weighted_unifrac()`

Examples

```
variance_adjusted_unifrac(ex_counts, tree = ex_tree)
```

wave_hedges *Wave Hedges distance*

Description

A distance metric derived from the Hedges' distance.

Usage

```
wave_hedges(
  counts,
  margin = 1L,
  norm = "none",
  pseudocount = NULL,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
norm	Normalize the incoming counts. Options are: • 'none': No transformation. • 'percent': Relative abundance (sample abundances sum to 1). • 'binary': Unweighted presence/absence (each count is either 0 or 1). • 'clr': Centered log ratio. • 'rclr': Robust centered log ratio. Default: 'none'.
pseudocount	Value added to counts to handle zeros when norm = 'clr'. Ignored for other normalization methods. See **Pseudocount** section.
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Wave Hedges distance is defined as:

$$\sum_{i=1}^{n} \frac{|X_i - Y_i|}{\max(X_i, Y_i)}$$

Where:

- X_i, Y_i : Absolute abundances of the i-th feature.
- n : The number of features.

Base R Equivalent:

```
x <- ex_counts[1,]
y <- ex_counts[2,]
sum(abs(x - y) / pmax(x, y))
```

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

Pseudocount

The `pseudocount` parameter is only relevant when `norm = 'clr'`.

Zeros are undefined in the centered log-ratio (CLR) transformation. If `norm = 'clr'`, `pseudocount` is `NULL` (the default), and zeros are detected, the function uses half the minimum non-zero value (`min(x[x>0]) / 2`) and issues a warning.

To suppress the warning, provide an explicit value (e.g., `1`).

Why this matters: The choice of pseudocount is not neutral; it acts as a weighting factor that can significantly distort downstream results, especially for sparse datasets. See Gloor et al. (2017) and Kaul et al. (2017) for open-access discussions on the mathematical implications, or Costea et al. (2014) for the impact on community clustering.

See `aitchison` for references.

References

Cha, S.-H. (2007). Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 1(4), 300–307.

See Also

beta_div(), vignette('bdiv'), vignette('bdiv_guide')

Other Abundance metrics: aitchison(), bhattacharyya(), bray(), canberra(), chebyshev(), chord(), clark(), divergence(), euclidean(), gower(), hellinger(), horn(), jensen(), jsd(), lorentzian(), manhattan(), matusita(), minkowski(), morisita(), motyka(), psym_chisq(), robust_aitchison(), soergel(), squared_chisq(), squared_chord(), squared_euclidean(), topsoe()

Examples

```
wave_hedges(ex_counts)
```

---

weighted_unifrac *Weighted UniFrac*

Description

A phylogenetic distance metric that accounts for the relative abundance of lineages.

Usage

```
weighted_unifrac(
  counts,
  tree = NULL,
  margin = 1L,
  pairs = NULL,
  cpus = n_cpus()
)
```

Arguments

counts	A numeric matrix of count data (samples × features). Typically contains absolute abundances (integer counts), though proportions are also accepted.
tree	A phylo-class object representing the phylogenetic tree for the OTUs in counts. The OTU identifiers given by colnames(counts) must be present in tree. Can be omitted if a tree is embedded with the counts object or as attr(counts, 'tree').
margin	The margin containing samples. 1 if samples are rows, 2 if samples are columns. Ignored when counts is a special object class (e.g. phyloseq). Default: 1
pairs	Which combinations of samples should distances be calculated for? The default value (NULL) calculates all-vs-all. Provide a numeric or logical vector specifying positions in the distance matrix to calculate. See examples.
cpus	How many parallel processing threads should be used. The default, n_cpus(), will use all logical CPU cores.

Details

The Weighted UniFrac distance is defined as:

$$\sum_{i=1}^{n} L_i |P_i - Q_i|$$

Where:

- n : The number of branches in the tree.
- L_i : The length of the i-th branch.
- P_i, Q_i : The proportion of the community descending from branch i in sample P and Q.

Input Types

The `counts` parameter is designed to accept a simple numeric matrix, but seamlessly supports objects from the following biological data packages:

- `phyloseq`
- `rbiom`
- `SummarizedExperiment`
- `TreeSummarizedExperiment`

For large datasets, standard matrix operations may be slow. See `vignette('performance')` for details on using optimized formats (e.g. sparse matrices) and parallel processing.

References

Lozupone, C. A., Hamady, M., Kelley, S. T., & Knight, R. (2007). Quantitative and qualitative beta diversity measures lead to different insights into factors that structure microbial communities. *Applied and Environmental Microbiology*, 73(5), 1576-1585. doi:10.1128/AEM.0199606

See Also

`beta_div()`, `vignette('bdiv')`, `vignette('bdiv_guide')`

Other Phylogenetic metrics: `faith()`, `generalized_unifrac()`, `normalized_unifrac()`, `unweighted_unifrac()`, `variance_adjusted_unifrac()`

Examples

```
weighted_unifrac(ex_counts, tree = ex_tree)
```

Index

∗ **Abundance metrics**
aitchison, 5
bhattacharyya, 14
bray, 15
canberra, 19
chebyshev, 22
chord, 24
clark, 26
divergence, 28
euclidean, 30
gower, 38
hellinger, 42
horn, 43
jensen, 48
jsd, 50
lorentzian, 54
manhattan, 56
matusita, 59
minkowski, 63
morisita, 66
motyka, 67
psym_chisq, 74
robust_aitchison, 79
soergel, 83
squared_chisq, 87
squared_chord, 88
squared_euclidean, 90
topsoe, 94
wave_hedges, 99
∗ **Diversity metrics**
brillouin, 18
fisher, 35
inv_simpson, 45
shannon, 80
simpson, 82
∗ **Dominance metrics**
berger, 9
mcintosh, 61
∗ **Phylogenetic metrics**
faith, 34
generalized_unifrac, 36
normalized_unifrac, 70
unweighted_unifrac, 95
variance_adjusted_unifrac, 97
weighted_unifrac, 101
∗ **Presence/Absence metrics**
hamming, 40
jaccard, 47
ochiai, 73
sorensen, 85
∗ **Richness metrics**
ace, 3
chao1, 21
margalef, 58
menhinick, 62
observed, 72
squares, 92
∗ **datasets**
ex_counts, 32
ex_tree, 33

ace, 3, *22*, *59*, *63*, *72*, *93*
aitchison, 5, *13*, *15*, *17*, *20, 21*, *24*, *26–28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *57, 58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90–92*, *95*, *100, 101*
alpha_div, 7

berger, 9, *62*
beta_div, 10
bhattacharyya, *6*, 14, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*
bray, *6*, 15, *15*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*
brillouin, 18, *36*, *46*, *82, 83*

canberra, *6*, *15*, *17*, 19, *24*, *26*, *28*, *30*, *32*, *40*,

43, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*
chao1, *4*, 21, *59*, *63*, *72*, *93*
chebyshev, *6*, *15*, *17*, *21*, 22, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*
chord, *6*, *15*, *17*, *21*, 24, *24*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*
clark, *6*, *15*, *17*, *21*, *24*, 26, *26*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*

divergence, *6*, *15*, *17*, *21*, *24*, *26*, 28, *28*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*

euclidean, *6*, *15*, *17*, *21*, *24*, *26*, *28*, 30, *30*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*
ex_counts, 32
ex_tree, 33

faith, 34, *38*, *71*, *97, 98*, *102*
fisher, *19*, 35, *46*, *82, 83*

generalized_unifrac, *35*, 36, *71*, *97, 98*, *102*
gower, *6*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, 38, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*

hamming, 40, *48*, *74*, *87*
hellinger, *6*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, 42, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*
horn, *6*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, 43, *43*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*

inv_simpson, *19*, *36*, 45, *82, 83*

jaccard, *42*, 47, *74*, *87*
jensen, *6*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, 48, *51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*
jsd, *6*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, 50, *50*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*

list_metrics, 51
lorentzian, *6*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, 54, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*

manhattan, *7*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, 56, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*
margalef, *4*, *22*, 58, *63*, *72*, *93*
match_metric *(*list_metrics*)*, 51
matusita, *7*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, 59, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*
mcintosh, *10*, 61
menhinick, *4*, *22*, *59*, 62, *72*, *93*
minkowski, *7*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, 63, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*
morisita, *7*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, 66, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*
motyka, *7*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, 67, *67*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, *101*

n_cpus, 69
normalized_unifrac, *35*, *38*, 70, *97, 98*, *102*

observed, *4*, *22*, *59*, *63*, 72, *93*
ochiai, *42*, *48*, 73, *87*

psym_chisq, *7*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, 74, *80*, *85*, *88*, *90*, *92*, *95*, *101*

rarefy, 76
read_tree, 78
robust_aitchison, *7*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, 79, *85*, *88*, *90*, *92*, *95*, *101*

shannon, *19*, *36*, *46*, 80, *83*
simpson, *19*, *36*, *46*, 82, *82*
soergel, *7*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, 83, *88*, *90*, *92*, *95*, *101*
sorensen, *42*, *48*, *74*, 85
squared_chisq, *7*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, 87, *90*, *92*, *95*, *101*

squared_chord, *7*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, 88, *88*, *92*, *95*, *101*
squared_euclidean, *7*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, 90, *90*, *95*, *101*
squares, *4*, *22*, *59*, *63*, *72*, 92

topsoe, *7*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, 94, *101*

unweighted_unifrac, *35*, *38*, *71*, 95, *98*, *102*

variance_adjusted_unifrac, *35*, *38*, *71*, 97, *97*, *102*

wave_hedges, *7*, *15*, *17*, *21*, *24*, *26*, *28*, *30*, *32*, *40*, *43*, *45*, *50, 51*, *55*, *58*, *60*, *65*, *67*, *69*, *75*, *80*, *85*, *88*, *90*, *92*, *95*, 99
weighted_unifrac, *35*, *38*, *71*, *97, 98*, 101