Package ‘sprex’

May 9, 2026

Type Package

Title Species Richness and Extrapolation

Description Functions for calculating species richness for rarefaction and extrapolation, primarily non-parametric species richness such as jackknife, Chao1, and ACE. Also available are functions for plotting species richness and extrapolation curves, and computing standard diversity and entropy indices.

Version 1.4.3

URL `https://github.com/SWFSC/sprex`

BugReports `https://github.com/SWFSC/sprex/issues`

Depends R (>= 4.1.0)

Imports ggplot2, stats, swfscMisc (>= 1.4)

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Eric Archer [aut, cre]

Maintainer Eric Archer <`eric.ivan.archer@gmail.com`>

Repository CRAN

Date/Publication 2025-06-10 13:00:02 UTC

Contents

`sprex-package` *Calculate species richness and extrapolation metrics*

Description

Calculate species richness and expected number of species primarily based on algorithms in Colwell, R.K., A. Chao, N.J. Gotelli, S.-Y. Lin, C.X. Mao, R.L. Chazdon, and J.T. Longino. 2012. Models and estimators linking individual-based and sample-based rarefaction, extrapolation and comparison of assemblages. Journal of Plant Ecology 5(1):3-21.

Details

sprex

"_PACKAGE"

`ACE` *Number of Unobserved Species*

Description

Calculate the number of unobserved species (f0).

Usage

```
ACE(f)

Chao1(f)

Clench(f, pct.n = 0.85, num.reps = 100)

Swor1(f, N)

iChao1(f)

jack1(f)

jack2(f)
```

Arguments

`f`	a vector of species frequencies where `f[i]` is the number of species represented by only `i` samples.
`pct.n`	percent of samples to use in bootstrap draws. Must be in range of `0:1`.
`num.reps`	number of random re-orderings of samples to fit curve to.
`N`	population size.

Value

All functions return a vector containing the estimated number of species (`s.est`), unobserved species (`f0`), observed species (`s.obs`), and the total number of samples (`n`). `Swor1` also returns the standard deviation of `s.est` as `sd.s.est`.

Author(s)

Eric Archer `<eric.archer@noaa.gov>`

References

`Chao1`, `ACE` Colwell, R.K., A. Chao, N.J. Gotelli, S.-Y. Lin, C.X. Mao, R.L. Chazdon, and J.T. Longino. 2012. Models and estimators linking individual-based and sample-based rarefaction, extrapolation and comparison of assemblages. Journal of Plant Ecology 5(1):3-21.

`jack1`, `jack2` Burnham, KP and WS Overton. 1978. Estimation of the size of a closed population when capture probabilities vary among animals. Biometrika 65(3):625-633.

`Swor1` Chao, A. and C.-W. Lin. 2012. Nonparametric lower bounds for species richness and shared species richness under sampling without replacement. Biometrics 68:912-921.

`iChao1` Chiu, C-H, Wang, Y-T, Walther, BA, and A Chao. 2014. An impro.ved nonparametric lower bound of species richness via a modified Good-Turing frequency formula. Biometrics 70(3):671-682.

`clench` Clench, H. 1979. How to make regional lists of butterflies: Some thoughts. Journal of the Lepidopterists' Society 33(4):216-231.

Examples

```
data(osa.second.growth)
f <- expand.freqs(osa.second.growth)

ace.est <- ACE(f)
chao1.est <- Chao1(f)
jack1.est <- jack1(f)
jack2.est <- jack2(f)
swor1.est <- Swor1(f, 20000)
ichao1.est <- iChao1(f)
clench.est <- Clench(f, num.reps = 50)

f0.est <- cbind(
  ACE = ace.est["f0"],
  Chao1 = chao1.est["f0"],
```

```
  jack1 = jack1.est["f0"],
  jack2 = jack2.est["f0"],
  Swor1 = swor1.est["f0"],
  iChao1 = ichao1.est["f0"],
  clench = clench.est["f0"]
)
f0.est
```

`bootstrap.assemblage` *Bootstrap Assemblage of Species*

Description

Create bootstrap assemblage of species.

Usage

```
bootstrap.assemblage(f, f0.func, n.boot = 500, ...)
```

Arguments

`f`	a vector of species frequencies where `f[i]` is the number of species represented by only `i` samples.
`f0.func`	function calculating the unobserved number of species (`f0`).
`n.boot`	number of bootstrap replicates.
`...`	other arguments to `f0.func`.

Value

a list of bootstrap replicates of species frequencies.

Author(s)

Eric Archer <`eric.archer@noaa.gov`>

References

Chao, A., N.J. Gotelli, T.C. Hsieh, E.L. Sander, K.H. Ma, R.K. Colwell, and A.M. Ellison. 2014. Rarefaction and extrapolation with Hill numbers: a framework for sampling and estimation in species diversity studies. Ecological Monographs 84(1):45-67.

`discovery.curve` *Discovery Curve*

Description

Calculate the components of a species discovery curve.

Usage

```
discovery.curve(
  f,
  f0.func,
  max.x = sum(f * 1:length(f)),
  n.pts = 100,
  ci = 0.95,
  plot = TRUE,
  ...
)
```

Arguments

`f`	a vector of species frequencies where `f[i]` is the number of species represented by only `i` samples.
`f0.func`	function to use to calculate `f0`.
`max.x`	the maximum number of samples to calculate the curve for. Defaults to the sample size of `f`.
`n.pts`	number of points between 0 and `max.x` to estimate.
`ci`	size of the confidence interval (0.5:1).
`plot`	plot the curve?
`...`	other arguments to `f0.func`.

Value

a list with:

`f.stats`	a named vector from `f0.func`.
`curve`	a `data.frame` defining the rarefaction and extrapolation curves (specified in the `section` column), and columns providing the lower (`lci`) and upper (`uci`).

Author(s)

Eric Archer <`eric.archer@noaa.gov`>

References

Colwell, R.K., A. Chao, N.J. Gotelli, S.-Y. Lin, C.X. Mao, R.L. Chazdon, and J.T. Longino. 2012. Models and estimators linking individual-based and sample-based rarefaction, extrapolation and comparison of assemblages. Journal of Plant Ecology 5(1):3-21.

Examples

```
data(osa.old.growth)
f <- expand.freqs(osa.old.growth)
d <- discovery.curve(f, f0.func = Chao1, max.x = 1200)

print(str(d))
```

`diversity` *Diversity Indices*

Description

Calculate common diversity and entropy indices.

Usage

```
diversity(
  x,
  type = c("effective.number", "richness", "shannon", "simpson", "gini.simpson",
   "unb.gini", "eveness.simpson", "eveness.pielou", "inv.simpson", "renyi", "hill"),
  q = NULL
)
```

Arguments

`x`	vector or matrix of values (character, factor) representing a class, from which proportions will be computed. If numeric, values will be converted to proportions. If a matrix, indices will be computed for all columns.
`type`	type of index to compute. See Details for descriptions. If `"renyi"` or `"hill"`, then `"q"` must be specified.
`q`	order of Hill number (must be >= 0).

Value

if a vector is supplied for `x`, a single value for the chosen type of index. If a matrix, a vector values for each column.

Note

Available indices for `type` are:

richness the number of observed classes (non-`NA` and frequency > 0)

effective.number exponent of Hill number of order 1

shannon Shannon entropy

simpson Simpson concentration

gini.simpson Gini-Simpson index (= 1 - Simpson concentration)

inv.simpson Inverse Simpson concentration

unb.gini unbiased Gini-Simpson index with correction for small sample sizes

eveness.simpson Simpson eveness

eveness.pielou Pielou eveness

renyi Renyi entropy

hill Hill number

Author(s)

Eric Archer <`eric.archer@noaa.gov`>

Examples

```
x <- sample(letters[1:4], 100, replace = TRUE, p = c(1, 2, 3, 4))

types <- c("richness", "effective.number", "shannon",
  "simpson", "inv.simpson", "gini.simpson", "unb.gini",
  "eveness.simpson", "eveness.pielou"
)

sapply(types, function(tp) diversity(x, type = tp))

# hill numbers with increasing order
order <- 0:5
hill.num <- sapply(order, function(q) diversity(x, type = "hill", q = q))
hill.num
plot(order, hill.num, type = "b")

# a matrix of frequencies
spp.freq <- cbind(
  sample(letters[1:4], 100, replace = TRUE, p = c(1, 1, 1, 4)),
  sample(letters[1:4], 100, replace = TRUE, p = c(4, 1, 1, 1)),
  sample(letters[1:4], 100, replace = TRUE, p = c(1, 1, 1, 1))
)

diversity(spp.freq, type = "eff")
```

`expand.freqs` *Expand Frequency Matrix*

Description

Expand a matrix or data.frame of species frequencies to full vector.

Usage

```
expand.freqs(freq.mat)
```

Arguments

`freq.mat`	a two column matrix or data.frame where the first column is the number of samples, and the second column is the number of species represented by with that many samples.

Value

a vector(`f`) of species frequencies where each element (`f[i]`) is the number of species represented by only `i` samples.

Author(s)

Eric Archer `<eric.archer@noaa.gov>`

Examples

```
data(osa.old.growth)
f <- expand.freqs(osa.old.growth)
f
```

`expected.num.species` *Expected Number of Species*

Description

Calculate the expected number of species for a given sample size.

Usage

```
expected.num.species(m, f, f0.func, ...)
```

Arguments

`m`	number of samples.
`f`	a vector of species frequencies where `f[i]` is the number of species represented by only `i` samples.
`f0.func`	a function that computes the number of unobserved species (f0).
`...`	other arguments to `f0.func`.

Value

a vector or matrix (depending on whether `m` is a scalar or vector, respectively) of the estimated number of species (`s.ind`) seen in `m` samples, and the standard deviation (`sd.s.ind`).

Author(s)

Eric Archer `<eric.archer@noaa.gov>`

References

Eqns 4, 5, 9, and 10 in Colwell, R.K., A. Chao, N.J. Gotelli, S.-Y. Lin, C.X. Mao, R.L. Chazdon, and J.T. Longino. 2012. Models and estimators linking individual-based and sample-based rarefaction, extrapolation and comparison of assemblages. Journal of Plant Ecology 5(1):3-21.

Examples

```
data(osa.old.growth)
f <- expand.freqs(osa.old.growth)
expected.num.species(60, f = f, f0.func = Chao1)

expected.num.species(c(60, 70, 75), f = f, f0.func = Chao1)
```

`f.stats` *Frequency Vector Statistics*

Description

Number of observed species and samples in species frequency vector.

Usage

```
f.stats(f)
```

Arguments

`f`	a vector of species frequencies where `f[i]` is the number of species represented by only `i` samples.

Value

a vector of the number of observed species (`s.obs`), and the total number of samples (`n`).

Author(s)

Eric Archer `<eric.archer@noaa.gov>`

Examples

```
data(osa.second.growth)
f <- expand.freqs(osa.second.growth)
f.stats(f)
```

`num.samples.required` *Number of Samples Required*

Description

Calculate the additional number of samples (individuals) to required to observe a given proportion of the total number of species.

Usage

```
num.samples.required(g, f, f0.func, ...)
```

Arguments

`g`	propotion of total number of species.
`f`	a vector of species frequencies where `f[i]` is the number of species represented by only `i` samples.
`f0.func`	a function that computes the number of unobserved species (f0).
`...`	other arguments to `f0.func`.

Value

a vector containing of the estimated additional number of samples (`m.g`) required to observe `g` percent of the total number of species.

Author(s)

Eric Archer `<eric.archer@noaa.gov>`

References

Eqn 12 in Chao, A., R.K. Colwell, C.-W. Lin, and N.J. Gotelli. 2009. Sufficient sampling for asymptotic minimum species richness estimators. Ecology 90(4):1125-1133.
Eqn 11 in Colwell, R.K., A. Chao, N.J. Gotelli, S.-Y. Lin, C.X. Mao, R.L. Chazdon, and J.T. Longino. 2012. Models and estimators linking individual-based and sample-based rarefaction, extrapolation and comparison of assemblages. Journal of Plant Ecology 5(1):3-21.

Examples

```
data(osa.old.growth)
f <- expand.freqs(osa.old.growth)
num.samples.required(0.6, f = f, f0.func = Chao1)
```

`osa.old.growth` *Osa beetle species counts*

Description

Matrices of the number of beetle species (`fi`) occuring `i` times in a survey.

Usage

```
data(osa.old.growth)
data(osa.second.growth)
```

References

Janzen DH (1973) Sweep samples of tropical foliage insects: effects of seasons, vegetation types, elevation, time of day, and insularity. Ecology 54:687-708.
Janzen DH (1973) Sweep samples of tropical foliage insects: description of study sites, with data on species abundances and size distributions. Ecology 54:659-86.

`rarefaction.overlap` *Rarefaction Overlap*

Description

Calculate the percent of overlap between two species estimate distributions where the larger sample size has been rarefied to match the smaller sample size.

Usage

```
rarefaction.overlap(x, y, f0.func, n.rare = NULL, ...)
```

Arguments

`x, y`	two vectors of species frequencies where the `i-th` element is the number of species represented by only `i` samples.
`f0.func`	function to use to calculate f0. Can be `Chao1`, `ACE`, `jack1`, `jack2`, `iChao1`, or `Swor1`.
`n.rare`	sample size to rarefy both populations to. Must be <= the minimum sample size. If `NULL`, the minimum sample size is used.
`...`	other arguments to `f0.func`.

Details

Calculates the expected number of species and the standard deviation for the smaller sample size of `x` and `y` using the frequency distributions of each. The function then fits a gamma distribution to each of these estimates, and returns the percent of overlap as the integral of the mininum value of the PDF for the two distributions. Integration takes place from 0 to the largest quantile representing 0.99999 of either distribution.

Value

a vector with the percent of overlap between the two distributions, the sample size, and species estimates for the `x` and `y` vectors.

Author(s)

Eric Archer <`eric.archer@noaa.gov`>

References

Colwell, R.K., A. Chao, N.J. Gotelli, S.-Y. Lin, C.X. Mao, R.L. Chazdon, and J.T. Longino. 2012. Models and estimators linking individual-based and sample-based rarefaction, extrapolation and comparison of assemblages. Journal of Plant Ecology 5(1):3-21.

See Also

`discovery.curve`

Examples

```
data(osa.old.growth)
data(osa.second.growth)
x <- expand.freqs(osa.old.growth)
y <- expand.freqs(osa.second.growth)
rarefaction.overlap(x, y, Chao1)
```

`sample.to.species.freq`

Create Vector of Species Frequencies

Description

Create vector of species frequencies from vector of sample frequencies.

Usage

```
sample.to.species.freq(x, min.f = NULL)
```

Arguments

`x`	a vector where `x[i]` is of the number of samples in the `i`-th species.
`min.f`	minimum size of return vector. Return vector is zero-padded up to this length if it would normally be shorter.

Value

a vector(`f`) of species frequencies where `f[i]` is the number of species represented by only `i` samples.

Author(s)

Eric Archer <`eric.archer@noaa.gov`>

See Also

species.to.sample.freq

Examples

```
x <- sample(1:20, 20, rep = TRUE)
f <- sample.to.species.freq(x)
print(x)
print(f)
```

`species.to.sample.freq`

Create Vector of Sample Frequencies

Description

Create vector of sample frequencies from vector of species frequencies.

Usage

```
species.to.sample.freq(f)
```

Arguments

`f`	a vector of species frequencies where `f[i]` is the number of species represented by only `i` samples.

Value

a vector(x) where `x[i]` is of the number of samples in the `i`-th species.

Author(s)

Eric Archer <`eric.archer@noaa.gov`>

See Also

sample.to.species.freq

Examples

```
data(osa.old.growth)
f <- expand.freqs(osa.old.growth)
x <- species.to.sample.freq(f)
print(f)
print(x)
```

Index