Package 'sads'

July 23, 2025

Type Package

Title Maximum Likelihood Models for Species Abundance Distributions

Version 0.6.3

Date 2024-01-16

Author Paulo I. Prado, Murilo Dantas Miranda and Andre Chalom

Maintainer Paulo I. Prado <prado@ib.usp.br>

Depends R (>= 3.5.0), stats, stats4, bbmle (>= 1.0.19)

Imports MASS, methods, graphics, grDevices, VGAM, poilog, GUILDS, poweRlaw

Suggests vegan

Description Maximum likelihood tools to fit and compare models of species abundance distributions and of species rank-abundance distributions.

License GPL-2

LazyData true

Encoding UTF-8

URL https://github.com/piLaboratory/sads

BugReports https://github.com/piLaboratory/sads/issues

RoxygenNote 7.2.3

NeedsCompilation yes

Repository CRAN

Date/Publication 2024-01-16 23:00:02 UTC

Contents

| sads-package | *Maximum Likelihood Models for Species Abundance Distributions* |

Description

Maximum likelihood tools to fit and compare models of species abundance distributions and of species rank-abundance distributions.

Details

The distribution of abundances of species is one of the basic patterns of ecological communities. The empirical distributions of abundances (SADs) or their ranks (RADs) are traditionally modelled through probability distributions. Hence, the maximum likelihood method can be used to fit and compare competing models for SADs and RADs. The **sads** package provides functions, classes and methods to:

- Fit classic SAD models: log-series, lognormal, broken-stick, . . . ;
- Fit classic rank-abundance (RADs) models: geometric, broken-stick, Zipf, Zipf-Mandelbrodt, . . . ;
- Tools for quick diagnostic and comparison of models;
- Tools to simulate Poisson and Negative Binomial samples from abundances in communities.

Author(s)

Paulo I. Prado, Murilo Dantas Miranda and Andre Chalom

Maintainer: Paulo I. Prado <prado@ib.usp.br>

References

Magurran, A.E. 2004. *Measuring Biological Diversity*. Blackwell.

Magurran, A.E. and McGill, B.J. 2011. *Biological Diversity – Frontiers in measurement and assessment*. Oxford University Press.

May, R.M. 1975. Patterns of Species Abundance and Diversity. In M. L. Cody and J. M. Diamond (Eds.), (pp. 81–120). Harvard University Press.

Green,J. and Plotkin, J.B. 2007 A statistical theory for sampling species abundances. *Ecology Letters 10*:1037–1045.

Saether, B.E., Engen, S. and Grotan, V. 2013. Species diversity and community similarity in fluctuating environments: parametric approaches using species abundance distributions. *Journal of Animal Ecology, 82*(4): 721–738.

See Also

vignettes of **sads**; vegan-package and poilog-package

Examples

```
## Rank-abundance plot
plot( rad(moths) )
## Preston's plots
plot (octav(moths) )
## Fit logseries model
moths.ls <-  fitsad(moths, sad = 'ls')
## Diagnostic plots
par(mfrow=c(2,2))
plot(moths.ls)
par(mfrow = c(1,1))
## Model summary
summary(moths.ls)
## Confidence interval
confint(moths.ls)
```

ARN82   *Biomass of marine animals in colonizing experiments in Baltic Sea*

Description

A numeric vector of biomass of species of benthonic marine invertebrates that colonized experimental containers laid in the Baltic Sea.

Usage

```
data(ARN82.eB.apr77)
```

Format

Named vector of positive numbers. Labels are a sequential numeric code for species, in the same sequence as the original table.

Details

Each value is the biomass of invertebrate species sampled in 13 containers laid in the seafloor,as reported in Table I in Arntz & Rumohr (1982). After the first year of exposition, 13 containers were recovered every two months, starting in December 1976. Data are from containers recovered in April 1977.

Source

Arntz, W. E., & Rumohr, H. (1982). An experimental study of macrobenthic colonization and succession, and the importance of seasonal variation in temperate latitudes. Journal of experimental marine biology and ecology, 64(1), 17-45.

References

Hughes, R. (1986). Theories and models of species abundance. The American Naturalist, 128(6), 879-899.

| bci | *Tree species abundance in Barro Colorado Island Plot* |

Description

Number of species of trees with diameter at breast height >= 10 mm in a 50 ha plot of tropical forest in the Barro Colorado Island, Panama.

Usage

```
data(bci)
```

Format

A named vector of 225 integers.

Details

The BCI forest dynamics research project was made possible by National Science Foundation grants to Stephen P. Hubbell: DEB-0640386, DEB-0425651, DEB-0346488, DEB-0129874, DEB-00753102, DEB-9909347, DEB-9615226, DEB-9615226, DEB-9405933, DEB-9221033, DEB-9100058, DEB-8906869, DEB-8605042, DEB-8206992, DEB-7922197, support from the Center for Tropical Forest Science, the Smithsonian Tropical Research Institute, the John D. and Catherine T. MacArthur Foundation, the Mellon Foundation, the Small World Institute Fund, and numerous private individuals, and through the hard work of over 100 people from 10 countries over the past two decades. The plot project is part the Center for Tropical Forest Science, a global network of large-scale demographic tree plots.

Source

Datasheet 'full matrix' of supplemental table 1 in Condit et al. (2002).

References

Condit et. al 2002. Beta-diversity in tropical forest trees. *Science 295*:666-669

Hubbell, S.P., Condit, R., and Foster, R.B. 2005. Barro Colorado Forest Census Plot Data. URL https://ctfs.arnarb.harvard.edu/webatlas/datasets/bci.

Condit, R. 1998. *Tropical Forest Census Plots*. Springer-Verlag and R. G. Landes Company, Berlin, Germany, and Georgetown, Texas.

---

birds   *Abundances of breeding birds in Quacker Run Valley, NY*

Description

Number of sights of nesting birds by species in a census in a 16,697 acres Park in New York State, ran in 1936 by A. A. Saunders.

Usage

```
data(birds)
```

Format

An unnamed vector of integers

Details

This is one of the datasets that F. W. Preston used in order to show the application of the lognormal distribution to describe the abundance of species in a sample of a biological community.

Source

Table IA of Preston (1948), which did not provide species' names.

References

Preston, F. W. 1948. The Commonness, and rarity, of species. *Ecology 29*:254-283.

Saunders, A. A. 1936. Studies of breeding birds in the Allegany State Park. New York State Museum Handbook 16. Albany, NY.

---

coefficients,fitsad-method
                                Standard stats methods

Description

Provide the standard interface for fitted objects

Usage

```
## S4 method for signature 'fitsad'
coefficients(object, ...)

## S4 method for signature 'fitsadC'
coefficients(object, ...)

## S4 method for signature 'fitrad'
coefficients(object, ...)

## S4 method for signature 'fitsad'
fitted.values(object, ...)

## S4 method for signature 'fitsadC'
fitted.values(object, ...)

## S4 method for signature 'fitrad'
fitted.values(object, ...)

## S4 method for signature 'fitsad'
fitted(object, ...)

## S4 method for signature 'fitsadC'
fitted(object, ...)

## S4 method for signature 'fitrad'
fitted(object, ...)

## S4 method for signature 'fitsad'
residuals(object, ...)

## S4 method for signature 'fitsadC'
residuals(object, ...)

## S4 method for signature 'fitrad'
residuals(object, ...)
```

Arguments

object	An object from class fitsad, fitrad or fitsadC
...	Other arguments to be forwarded for the lower level function

Details

These methods are provided to allow for standard manipulation of fitsad, fitsadC and fitrad objects using the generic methods defined in the "stats" package. Please see the original man pages for each method.

coefficients is an alias to coef (implemented in package "bbmle").

fitted and fitted.values provide an alternative interface to radpred; these are also used to calcutate residuals.

Notice that radpred is a preferred interface for most calculations, specially if there are several ties.

---

coverpred-class   *Class* "coverpred" *for predicted values for abundance classes*

Description

A list with values that define abundance classes and number and densities of species in each abundance, as predicted by a model of species abundance distribution.

Usage

```
## S4 method for signature 'coverpred'
points(x, y.scale = c("density", "Freq", "prob"), mid = TRUE, ...)
```

Arguments

x	an object of class coverpred
y.scale	"density" plots points in density scale, "Freq" plots frequencies and "prob" plots relative frequencies.
mid	logical; if TRUE x coordinates of abundances are set to the mid of each class, if FALSE x coordinates of abundances are the values of class upper limit.
...	further parameters to be passed to lines.

Objects from the Class

Objects can be created by calls of the form new("coverpred", ...), but most often by a call to coverpred.

Slots

.Data: Object of class "list" with five vectors: breaks of abundance classes, midpoints of abundance classes, upper limit of abundance classes, predicted proportion of species in each abundance class, predicted number of species in each abundance class.

Extends

Class "list", directly. Class "oldClass", by class "list", distance 2. Class "vector", by class "list", distance 3.

Methods

points signature(x = "coverpred"): adds frequency (or density) data contained in the object as points in a histogram.

Author(s)

Andre Chalom and Paulo I Prado <prado@ib.usp.br>

See Also

[coverpred](#) to get an object of the class from a histogran, [fitsadC](#) for fitting species abundance distributions to abundance data aggregated in classes.

Examples

```
## Example of fitting a sad model to cover data
## Abundance classes: cover scale for plants
Lbrk <- c(0,1,3,5,15,25,35,45,55,65,75,85,95,100)
## To fit a sad model to cover data, data sould be in histogram format
grass.h <- hist(grasslands$mids, breaks = Lbrk, plot = FALSE)
## Fits a Pareto distribution to the histogram object
grass.p <- fitparetoC(grass.h)
## Values (densities, frequencies, relative frenquecies) predicted by the model for each size class
grass.p.pred <- coverpred(grass.p)
## Plot histogram of observed values in density scale
plot(grass.h)
## adds points for the predicted values (predicted densities)
points(grass.p.pred)
```

---

coverpred-methods	*~~ Methods for Function* coverpred *in Package* **sads** *~~*

Description

Creates an object of `coverpred-class` with the number of species in each abundance class predicted by a species abundance distribution.

Arguments

object	an object of class `fitsadC` or `histogram`; fitted model of species abundances distributions. Alternatively a histogram with number of species in each abundance class.
sad	character; root name of sad distribution to calculate expected percentiles. See [fitsadC](#) for available distributions.
coef	named list of numeric values; parameter values of the distribution given in sad. Parameters should be named as in the corresponding density function, and in the same order.
trunc	non-negative integer, trunc > min(x); truncation point if fitted distribution is truncated.
breaks	real vector; breakpoints of abundance classes.
mids	real vector; breakpoints of abundance classes.
S	integer; total number of species.

Methods

signature(object = "fitsadC", sad = "missing", coef = "missing", trunc = "missing", breaks = "missing", mi
    number of species in each abundance class predicted from a sads model fitted with function
    fitsadC.

signature(object = "histogram", sad = "character", coef = "list", trunc = "ANY", breaks = "missing", mids =
    number of species in each abundance class predicted from abundance distribution named by
    sad with parameters defined in coef. Number of species S and intervals of the abundance
    classes defined are given by histogram.

signature(object = "missing", sad = "character", coef = "list", trunc = "ANY", breaks = "numeric", mids = "A
    number of species each abundance class predicted from abundance distribution named by sad
    with parameters defined in coef, number of species S defined in S. Abundance classes are
    defined by their breakpoints (breaks) or by their midpoints (mids).

Author(s)

Paulo I. Prado <prado@ib.usp.br>

Examples

```
## Example of fitting a sad model to cover data
## Abundance classes: cover scale for plants
Lbrk <- c(0,1,3,5,15,25,35,45,55,65,75,85,95,100)
## To fit a sad model to cover data, data sould be in histogram format
grass.h <- hist(grasslands$mids, breaks = Lbrk, plot = FALSE)
## Fits a Pareto distribution to the histogram object
grass.p <- fitparetoC(grass.h)
## Values (densities, frequencies, relative frenquecies) predicted by the model for each size class
grass.p.pred <- coverpred(grass.p)
## Plot histogram of observed values in density scale
plot(grass.h)
## adds points for the predicted values (predicted densities)
points(grass.p.pred)
## Predicted values for the same data but other parameter values
grass.p.pred2 <- coverpred(grass.h, sad = "pareto", coef = list(shape = 1, scale = 0.5))
## Adds the new predicted values to the plot
points(grass.p.pred2, col = "red")
```

---

dbs   *MacArthur's Broken-stick distribution*

Description

Density, distribution function, quantile function and random generation for the Broken-stick distri-
bution with parameters N and S.

Usage

```
dbs( x, N, S, log = FALSE )
pbs( q, N, S, lower.tail = TRUE, log.p = FALSE )
qbs( p, N, S, lower.tail = TRUE, log.p = FALSE )
rbs( n, N, S )
drbs( x, N, S, log = FALSE )
prbs( q, N, S, lower.tail = TRUE, log.p = FALSE )
qrbs( p, N, S, lower.tail = TRUE, log.p = FALSE )
rrbs( n, N, S)
```

Arguments

x	vector of (non-negative integer) quantiles. In the context of species abundance distributions, this is a vector of abundances (for dbs) or abundance ranks (for drbs) of species in a sample.
q	vector of (non-negative integer) quantiles. In the context of species abundance distributions, a vector of abundances (for dbs) or abundance ranks (for drbs) of species in a sample.
n	number of random values to return.
p	vector of probabilities.
N	positive integer $0 < N < \text{Inf}$, sample size. In the context of species abundance distributions, the sum of abundances of individuals in a sample.
S	positive integer $0 < S < \text{Inf}$, number of elements in a collection. In the context of species abundance distributions, the number of species in a sample.
log, log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].

Details

The Broken-stick distribution was proposed as a model for the expected abundance of elements in a collection:

$$n(i) = \frac{N}{S} \sum_{k=i}^{S} 1/k$$

where n(i) is the abundance in the i-th most abundant element (MacArthur 1960, May 1975). Hence the probability (or expected proportion of occurrences) in the i-th element is

$$p(i) = \frac{n(i)}{S} = S^{-1} \sum_{k=i}^{S} 1/k$$

[dpq]rbs stands for "rank-abundance Broken-stick" and return probabilities and quantiles based on the expression above, for p(i). Therefore, [dpq]rbs can be used as a rank-abundance model for species' ranks in a sample or in a biological community see `fitrad`.

The probability density for a given abundance value in the Broken-stick model is given by

$$p(x) = \frac{S-1}{N} \left(1 - \frac{x}{N} \right)^{S-2}$$

Where x is the abundance of a given element in the collection (May 1975). [dpq]bs return probabilities and quantiles according to the expression above for p(x). Therefore, [dpq]bs can be used as a species abundance model see `fitsad`.

Value

dbs gives the (log) density and pbs gives the (log) distribution function of abundances, and qbs gives the corresponding quantile function. drbs gives the (log) density and prbs gives the (log) distribution function of ranks, and qrbs gives the corresponding quantile function.

Author(s)

Paulo I Prado <prado@ib.usp.br> and Murilo Dantas Miranda.

References

MacArthur, R.H. 1960. On the relative abundance of species. *Am Nat 94*:25–36.

May, R.M. 1975. Patterns of Species Abundance and Diversity. In Cody, M.L. and Diamond, J.M. (Eds) *Ecology and Evolution of Communities*. Harvard University Press. pp 81–120.

See Also

`fitbs` and `fitrbs` to fit the Broken-stick distribution as a abundance (SAD) and rank-abundance (RAD) model.

Examples

```
x <- 1:25
PDF <- drbs(x=x, N=100, S=25)
CDF <- prbs(q=x, N=100, S=25)
par(mfrow=c(1,2))
plot(x,CDF, ylab="Cumulative Probability", type="b",
     main="Broken-stick rank distribution, CDF")
plot(x,PDF, ylab="Probability", type="h",
     main="Broken-stick rank distribution, PDF")
par(mfrow=c(1,1))

## quantile is the inverse of CDF
all.equal( qrbs( CDF, N=100, S=25), x) # should be TRUE
```

---

dgs   *Geometric series distribution*

Description

Density, distribution function, quantile function and random generation for the Geometric Series distribution, with parameter k.

Usage

```
dgs( x, k, S, log = FALSE )
pgs( q, k, S, lower.tail = TRUE, log.p = FALSE )
qgs( p, k, S, lower.tail = TRUE, log.p = FALSE )
rgs( n, k, S )
```

Arguments

x	vector of (non-negative integer) quantiles. In the context of species abundance distributions, this is a vector of abundance ranks of species in a sample.
n	number of random values to return.
k	positive real, $0 < k < 1$; geometric series coefficient; the ratio between the abundances of i-th and (i+1)-th species.
q	vector of (non-negative integer) quantiles. In the context of species abundance distributions, a vector of abundance ranks of species in a sample.
p	vector of probabilities.
S	positive integer $0 < S < Inf$, number of elements in a collection. In the context of species abundance distributions, the number of species in a sample.
log, log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].

Details

The Geometric series distribution gives the probability (or expected proportion of occurrences) of the i-th most abundant element in a collection:

$$p(i) = Ck(1 - k)^{i-1}$$

where C is a normalization constant which makes the summation of p(i) over S equals to one:

$$C = \frac{1}{1 - (1 - k)^S}$$

where S is the number of species in the sample.

Therefore, [dpq]gs can be used as rank-abundance model for species ranks in a sample or biological community see fitrad-class.

Value

dgs gives the (log) density and pgs gives the (log) distribution function of ranks, and qgs gives the corresponding quantile function.

Note

The Geometric series is NOT the same as geometric distribution. In the context of community ecology, the first can be used as a rank-abundance model and the former as a species-abundance model. See fitsad and fitrad and vignettes of **sads** package.

Author(s)

Paulo I Prado <prado@ib.usp.br> and Murilo Dantas Miranda.

References

Doi, H. and Mori, T. 2012. The discovery of species-abundance distribution in an ecological community. *Oikos 122:* 179–182.

May, R.M. 1975. Patterns of Species Abundance and Diversity. In Cody, M.L. and Diamond, J.M. (Eds) *Ecology and Evolution of Communities*. Harvard University Press. pp 81–120.

See Also

fitgs, fitrad to fit the Geometric series as a rank-abundance model.

Examples

```
x <- 1:25
PDF <- dgs(x=x, k=0.1, S=25)
CDF <- pgs(q=x, k=0.1, S=25)
par(mfrow=c(1,2))
plot(x,CDF, ylab="Cumulative Probability", type="b",
     main="Geometric series distribution, CDF")
plot(x,PDF, ylab="Probability, log-scale", type="h",
     main="Geometric series distribution, PDF", log="y")
par(mfrow=c(1,1))

## quantile is the inverse of CDF
all.equal(qgs(CDF, k=0.1, S=25), x)
```

---

distr   *Continuous or discrete distributions*

Description

Checks if the distribution is continuous or discrete

Usage

```
distr(distribution)
```

Arguments

distribution   Character. The name of the distribution ("geom" for "fitgeom", "weibull" for "fitweibull", etc.

Details

Returns whether a given distribution (used for a sad or rad model) is "discrete" or "continuous". The name is compared to a list of known distributions, so distributions not used in the sads package will return "NA" with a warning.

In the package sads up to version 0.2.3, the user was required to explicitly set a distr argument in some calls to radpred and qqsad. Now this is handled automatically, and attempts to set the "distr" argument explicitly are ignored.

---

dls	*Fisher's Log-series distribution*

Description

Density, distribution function, quantile function and random generation for the Fisher's log-series probability distribution with parameter alpha.

Usage

```
dls( x, N , alpha, log=FALSE)
pls(q, N, alpha, lower.tail=TRUE, log.p=FALSE)
qls(p, N, alpha, lower.tail = TRUE, log.p = FALSE)
rls(n, N, alpha)
```

Arguments

x	vector of (integer, x>0) quantiles. Usually a vector of abundances of species in a sample.
q	vector of (integer, x>0) quantiles. Usually a vector of abundances of species in a sample.
p	vector of probabilities.
n	number of random values to return.
N	sample size. Usually the total number of individuals in the sample (see details).
alpha	real positive; Fisher's alpha parameter (see details).
log, log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].

Details

The Fisher log-series is a limiting case of the Negative Binomial where the dispersion parameter of the negative binomial tends to zero. It was originally proposed by Fisher (1943) to relate the expected number of species in a sample from a biological community to the sample size as:

Where alpha is the single parameter of the log-series distribution, often used as a diversity index. From this relation follows that the expected number of species with x individuals in the sample is

$$S(x) = \alpha \frac{X^x}{x}$$

Where X is a function of alpha and N, that tends to one as the sample size N increases:

$$X = \frac{N}{\alpha + N}$$

The density function used here is derived by Alonso et al. (2008, supplementary material). In ecology, this density distribution gives the probability that a species has an abundance of x individuals in a random sample of size N of the community. In the community, the species abundances are independent random variables that follow a log-series distribution. Thus, a random sample of a log-series is also a log-series distribution.

Therefore, a log-series distribution is a model for species abundances distributions (SAD) under the assumptions that (a) species abundances in the community are independent identically distributed log-series variables, (b) sampling is a Poisson process, (c) sampling is done with replacement, or the fraction sampled is small enough to approximate a sample with replacement.

Value

`dls` gives the (log) of the density, `pls` gives the (log) distribution function, `qls` gives the (log) the quantile function.

Invalid values for parameter `alpha` will result in return values `NaN`, with a warning.

Author(s)

Paulo I Prado <prado@ib.usp.br> and Murilo Dantas Miranda.

References

Alonso, D. and Ostling, A., and Etienne, R. S. 2008 The implicit assumption of symmetry and the species abundance distribution. Ecology Letters, 11: 93-105.

Fisher, R.A, Corbert, A.S. and Williams, C.B. (1943) The Relation between the number of species and the number of individuals in a random sample of an animal population. The Journal of Animal Ecology, 12(1): 42–58.

Green,J. and Plotkin, J.B. 2007 A statistical theory for sampling species abundances. Ecology Letters 10:1037–1045

Pielou, E.C. 1977. Mathematical Ecology. New York: John Wiley and Sons.

See Also

dpois, dnbinom, [dpoig](#). For maximum likelihood estimation in the context of species abundance distributions see [fitls](#), fisherfit in **vegan** package and fisher in **untb** package.

Examples

```
x <- 1:100
PDF <- dls(x=x, N=100, alpha=5)
CDF <- pls(q=x, N=100, alpha=5)
par(mfrow=c(1,2))
plot(x,CDF, ylab="Cumulative Probability", type="b",
     main="Log-Series distribution, CDF")
plot(x,PDF, ylab="Probability", type="h",
     main="Log-Series distribution, PDF")
par(mfrow=c(1,1))

## Fisher log-series is a discrete PDF, hence:
all.equal(pls(10,N=1000,alpha=50), sum(dls(1:10,N=1000,alpha=50))) # should be TRUE

## qls is the inverse of pls
all.equal(qls(CDF,N=100,alpha=5), x) # should be TRUE
```

---

dmand   *Zipf-Mandelbrodt distribution*

Description

Density, distribution function, quantile function and random generation for Zipf-Mandelbrodt distribution with parameters N s and v.

Usage

```
dmand( x, N, s, v, log=FALSE)
pmand( q, N, s, v, lower.tail=TRUE, log.p=FALSE)
qmand( p, N, s, v, lower.tail = TRUE, log.p = FALSE)
rmand( n, N, s, v)
```

Arguments

x	vector of (non-negative integer) quantiles. In the context of species abundance distributions, this is a vector of abundance ranks of species in a sample.
q	vector of (non-negative integer) quantiles. In the context of species abundance distributions, a vector of abundance ranks of species in a sample.
n	number of random values to return.
p	vector of probabilities.
N	positive integer $0 < N < Inf$, total number of elements of a collection. In the context of species abundance distributions, usually the number of species in a sample.

s	positive real s > 0; Zipf-Mandelbrodt exponent.
v	positive real or zero v >= 0; Zipf-Mandelbrodt parameter.
log, log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].

Details

The Mandelbrodt distribution describes the probability or frequency of occurrence of a given element from a set of N elements. This probability is inversely proportional to a power s of the rank of the frequency of the element in the set. The density function is

$$p(x) = \frac{(x+v)^{-s}}{\sum_{i=1}^{N}(i+v)^{-s}}$$

Since p(x) is proportional to a power of x, the Zipf-Mandelbrodt distribution is a power distribution. The Zipf distribution is a special case when v=0. Hence, the Zipf-Mandelbrodt distribution is a generalization of the Zipf Law, and is widely used in the [x] for the same purposes. In Ecology, it can be used to describe the probability of the abundance rank x of given species in a sample or assemblage of N species.

Value

dmand gives the (log) density of the density, pmand gives the (log) distribution function, qmand gives the quantile function.

Invalid values for parameters v or s will result in return values NaN, with a warning.

Author(s)

Paulo I Prado <prado@ib.usp.br> and Murilo Dantas Miranda.

References

Johnson N.L., Kemp, A.W. and Kotz S. 2005. *Univariate Discrete Distributions*, 3rd edition, Hoboken, New Jersey: Wiley. Section 11.2.20.

Magurran, A.E. and McGill, B.J. 2011. *Biological Diversity - Frontiers in measurement and assessment*. Oxford: Oxford University Press.

See Also

dmand and rmand and related functions in **mandR** package; Zeta for zeta distribution in **VGAM** package.

Examples

```
x <- 1:100
PDF <- dmand(x=x, N=100, s=1.5, v=2)
CDF <- pmand(q=x, N=100, s=1.5, v=2)
par(mfrow=c(1,2))
plot(x,CDF, ylab="Cumulative Probability", type="b",
```

```
        main="Zipf-Mandelbrodt distribution, CDF")
plot(x,PDF, ylab="Probability", type="h",
        main="Zipf-Mandelbrodt distribution, PDF")
par(mfrow=c(1,1))

## quantile is the inverse of CDF
all.equal( qmand(p=CDF, N=100, s=1.5, v=2), x)

## Zipf distribution is a particular case of ZM when v=0
all.equal( dmand(x=x, N=100, s=1.5, v=0), dzipf(x=x, N=100, s=1.5) )
```

| dmzsm | *Metacommunity zero-sum multinomial distribution* |

Description

Density, distribution, quantile function and random generation for Alonso & McKane's mZSM distribution with parameter `theta`.

Usage

```
dmzsm(x, J, theta, log = FALSE)
pmzsm(q, J, theta, lower.tail=TRUE, log.p=FALSE)
qmzsm(p, J, theta, lower.tail = TRUE, log.p = FALSE)
rmzsm(n, J, theta)
```

Arguments

x	vector of (non-negative integer) quantiles. In the context of species abundance distributions, this is a vector of abundance of species in a sample.
q	vector of (non-negative integer) quantiles. In the context of species abundance distributions, a vector of abundance of species in a sample.
n	number of random values to return.
p	vector of probabilities.
J	positive integer $0 < J < \text{Inf}$, sample size. In the context of species abundance distributions, usually the number of individuals in a sample.
theta	positive real theta > 0; Hubbell's 'fundamental biodiversity number'
log, log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are $P[X <= x]$, otherwise, $P[X > x]$.

Details

The metacommunity Zero-sum multinomial distribution (mZSM) describes the probabilities of abundances (population sizes) in a random sample of size J taken from a collection of populations (the metacommunity). The total number of individuals in the metacommunity is fixed (zero-sum assumption). The populations in the metacommunity undergo a stochastic birth-death-immigration process, with equal demographic rates (neutrality or ecological equivalence assumption, Hubbell 2001). Alonso and McKane (2004) proposed an approximation for the density function for a large Poisson sample (J>100):

$$p(x) = \frac{N(x)}{\sum_1^S N(x)}$$

where S is the number of populations in the sample, and N(x) is the expected number of sampled populations of size x :

$$N(x) = \frac{\theta}{x(1 - x/J)^{\theta - 1}}$$

Therefore, the mZSM is a model for species abundances distributions (SAD) in a sample taken from a community under the assumptions that (a) species abundances in the community follows the stationary distribution of a neutral, zero-sum stochastic process of birth, death and speciation (or migration); (b) sampling is a Poisson process with expected value well approximated by N(x), (c) individuals are sampled with replacement, or the fraction of total individuals sampled is small enough to approximate a sample with replacement.

Value

dmzsm gives the (log) density function, pmzsm gives the (log) distribution function, and qmzsm gives the quantile function.

Invalid values for parameters J or theta will result in return values NaN, with a warning.

Author(s)

Paulo I Prado <prado@ib.usp.br>, Murilo Dantas Miranda and Andre Chalom

References

Alonso, D. and McKane, A.J. 2004. Sampling Hubbell's neutral model of biodiversity. *Ecology Letters* 7:901-910.

Hubbell, S.P. 2001. *The Unified Neutral Theory of Biodiversity*. Princeton University Press.

See Also

fitmzsm for maximum likelihood estimation; alonso.eqn12 in package **untb** which is based on the exact formulation of mZSM.

Examples

```
## Alonso & McKanne (2004) figure 2
data(moths) #Fisher's moths data
m.tab <- hist(moths, breaks = 2^(0:12), plot = FALSE)
plot(m.tab$density~m.tab$mids, log="xy",
     xlab = "Abundance", ylab = "Probability density",
     ylim=c(1e-7,1))
X <- 1:max(moths)
Y <- dmzsm(X, J = sum(moths), theta = 39.8)
lines(Y ~ X)
```

---

dpareto   *Pareto distribution*

Description

Density, distribution function, quantile function and random generation for the Pareto distribution with parameters shape and `scale`.

Usage

```
dpareto(x, shape, scale = min(x), log = FALSE)
ppareto(q, shape, scale = min(q), lower.tail = TRUE, log.p = FALSE)
qpareto(p, shape, scale = min(p), lower.tail = TRUE, log.p = FALSE)
rpareto(n, shape, scale = 1)
```

Arguments

x	vector of (non-negative integer) quantiles. In the context of species abundance distributions, this is a vector of abundances of species in a sample.
q	vector of (non-negative integer) quantiles. In the context of species abundance distributions, a vector of abundances of species in a sample.
n	number of random values to return.
p	vector of probabilities.
shape	positive real; shape parameter, a.k.a Pareto's index or tail index.
scale	positive real, scale >= min(x); scale parameter.
log, log.p	logical; if TRUE, probabilities p and densities d are given as log(p) and log(d).
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].

Details

The Pareto distribution is a continuous power-law density distribution with scale (a) and shape (b) parameters with the form:

$$f(x) = \frac{ba^b}{x^{b+1}}$$

For all x >= scale, and

f(x) = 0 otherwise.

The shape parameter is known as Pareto's index or tail index, and increases the decay of f(x). This distribution was originally used to describe the allocation of wealth or income among individuals in human societies. As a continuous counterpart of Zipf Law, Pareto distribution describes well many other variables that follow a power-law.

In ecology the Pareto distribution can be used to describe the distribution of abundances among species in a biological assemblage (a.k.a. biological community) or in a sample taken from such an assemblage. Though much less used than the lognormal to fit SADs, it can fit better the extremities of the empirical distributions to which the lognormal applies (Johnson et al. 1995, p.608).

Value

dpareto gives the (log) density, ppareto gives the (log) distribution function, qpareto gives the quantile function.

Invalid values for parameters shape or scale will result in return values NaN, with a warning.

Note

These functions implement the Pareto distribution of the first kind sensu Johnson et al. (1995, pp.574).

The pdf and cdf are defined as zero for all x < scale, but the functions [dp]pareto currently return an error if scale > min(x), to avoid some fitting and plotting problems.

Author(s)

Paulo I Prado <prado@ib.usp.br> and Murilo Dantas Miranda.

References

Johnson, N.L., Kotz, S. and Balakrishnan, N. 1995. *Continuous Univariate Distributions*, volume 2, chapter 20. Wiley, New York.

See Also

Pareto in packages **VGAM** and **actuar** for more general and flexible implementations; fitpareto for maximum likelihood estimation in the context of species abundance distributions.

Examples

```
par(mfrow=c(1,2))
curve(dpareto(x, shape=3, scale=1), 1,8, ylab="Density",
      main="Pareto PDF")
curve(ppareto(x, shape=3, scale=1), 1,8, ylab="Probability",
      main="Pareto CDF")
par(mfrow=c(1,1))


## Quantile is the inverse function of probability:
```

```
p.123 <-ppareto(1:3,shape=3,scale=0.99)
all.equal(qpareto(p.123, shape=3, scale=0.99), 1:3)
```

---

dpoig   *Compound Poisson-gamma distribution*

Description

Density, distribution function, quantile function and random generation for for the Poisson-gamma compound probability distribution with parameters `frac`, `rate` and `rate`.

Usage

```
dpoig(x, frac, rate, shape, log=FALSE)
ppoig(q, frac, rate, shape, lower.tail=TRUE, log.p=FALSE)
qpoig(p, frac, rate, shape, lower.tail=TRUE, log.p=FALSE)
rpoig(n, frac, rate, shape)
```

Arguments

x	vector of (non-negative integer) quantiles. In the context of species abundance distributions, this is a vector of abundances of species in a sample.
q	vector of (non-negative integer) quantiles. In the context of species abundance distributions, a vector of abundances of species in a sample.
n	number of random values to return.
p	vector of probabilities.
frac	single numeric '0<frac<=1'; fraction of the population or community sampled (see details)
rate	vector of (non-negative) rates of the gamma distribution of the sampled population (see details). Must be strictly positive.
shape	the shape parameter of the gamma distribution of the sampled population (see details). Must be positive.
log, log.p	logical; if TRUE, probabilities p are given as log(p)
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].

Details

A compound Poisson-gamma distribution is a Poisson probability distribution where its single parameter, the process mean rate, is frac*n, at which n is a random variable with gamma distribution. The density function is given by Green & Plotkin (2007).

In ecology, this distribution gives the probability that a species has an abundance of x individuals in a random sample of a fraction frac of the community. In the community the species abundances are independent random variables that follow a gamma density function.

Hence, a Poisson-gamma distribution is a model for species abundances distributions (SAD) under the assumptions: (a) species abundances in the community are independent identically distributed

gamma variables, (b) sampling is a Poisson process with expected value frac*n, (c) the sampling is done with replacement, or the fraction sampled is small enough to approximate a sample with replacement.

The Poisson-gamma distribution is also known as the Negative Binomial distribution. The function dpoig is provided to express the Negative Binomial explicitly as a compound distribution. The Fisher log-series (Fisher 1943) is a limiting case where the dispersion parameter of the Negative Binomial tends to zero. As in the case of the Poisson-exponential, the user should not fit to the Poisson-gamma directly, and should use instead the `fitls` function.

Value

(log) density of the (zero-truncated) density

Author(s)

Paulo I Prado <prado@ib.usp.br>, Cristiano Strieder and Andre Chalom.

References

Fisher, R.A, Corbert, A.S. and Williams, C.B. (1943) The Relation between the number of species and the number of individuals in a random sample of an animal population. *The Journal of Animal Ecology, 12*:42–58.

Green,J. and Plotkin, J.B. 2007 A statistical theory for sampling species abundances. *Ecology Letters 10*:1037–1045

Pielou, E.C. 1977. *Mathematical Ecology*. New York: John Wiley and Sons.

See Also

dgamma, dpois, dnbinom for related distributions, dpoix for the special case of Poisson-exponential

---

dpoilog	*Poisson-lognormal distribution*

Description

Density, distribution function, quantile function and random generation for Poisson-lognormal distribution with parameters mu and `sigma`.

Usage

```
dpoilog( x, mu, sig, log=FALSE)
ppoilog( q, mu, sig, lower.tail=TRUE, log.p=FALSE)
qpoilog( p, mu, sig, lower.tail = TRUE, log.p = FALSE)
rpoilog( n, mu, sig)
```

Arguments

x	vector of (non-negative integer) quantiles. Usually a vector of abundances of species in a sample.
q	vector of (non-negative integer) quantiles. Usually a vector of abundances of species in a sample.
n	number of random values to return.
p	vector of probabilities.
mu, sig	parameters of the compounding lognormal distribution (see details).
log, log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].

Details

A compound Poisson-lognormal distribution is a Poisson probability distribution where its single parameter lambda is a random variable with lognormal distribution. The density function is

$$p(x) = \frac{e^{x\mu + x^2\sigma/2}(2\pi\sigma)^{-1/2}}{x!} \, g(y)$$

where

$$g(y) = \int_{-\infty}^{\infty} e^{-e^y} \frac{e^{(-y-\mu-x\sigma)^2}}{2\sigma} \, dy$$

(Bulmer 1974 eq.5). For x = 0, 1, 2, ...

In ecology, this distribution gives the probability that a species has an abundance of x individuals in a random sample of a fraction 'f' of the community. In the community, the species abundances are independent random variables that follow a lognormal density function, with parameters (mu + ln(f), sigma) (Engen et al. 2002).

Hence, a Poisson-lognormal distribution is a model for species abundances distributions (SAD) in a sample taken from a community under the assumptions: (a) species abundances in the community are independent identically distributed lognormal variables, (b) sampling is a Poisson process with expected value E[x]= f*n where n is the abundance in the community and f the fraction of individuals sampled, (c) individuals are sampled with replacement, or the fraction of total individuals sampled is small enough to approximate a sample with replacement. See Engen (1977) and Alonso et al. (2008) for critical evaluations.

Value

'dpoilog' gives the (log) density of the density, 'ppoilog' gives the (log) distribution function, 'qpoilog' gives the quantile function.

Author(s)

Paulo I. Prado <prado@ib.usp.br>, Andre Chalom and Murilo Dantas Miranda

Source

These functions were built from dpoilog function from **poilog** package (Vidar Grøtan and Steinar Engen).

dpoilog is just a wrapper of poilog::dpoilog with an additional log argument.

ppoilog does the cumulative sum of poilog::dpoilog.

qpoilog uses modified bisection method to find numerically quantiles using ppoilog.

rpoilog selects random values from a poilog distribution. It is unrelated to the function of the same name in **poilog**.

References

Alonso, D. and Ostling, A., and Etienne, R. S. 2008 The implicit assumption of symmetry and the species abundance distribution. *Ecology Letters, 11*: 93-105.

Bulmer,M. G. 1974. On Fitting the Poisson Lognormal Distribution to Species-Abundance Data. *Biometrics, 30*: 101-110.

Grøtan V. and Engen S. 2008. poilog: Poisson lognormal and bivariate Poisson lognormal distribution. R package version 0.4.

Engen, S. 1977. Comments on two different approaches to the analysis of species frequency data. *Biometrics, 33*: 205-213.

Engen, S., R. Lande, T. Walla & P. J. DeVries. 2002. Analyzing spatial structure of communities using the two-dimensional Poisson lognormal species abundance model. *American Naturalist 160*: 60-73.

See Also

dpois, dlnorm; dpoilog and rpoilog in **poilog** package; rsad for random generation, fitpoilog for maximum likelihood estimation.

---

dpoix   *Compound Poisson-Exponential distribution*

Description

Density, distribution function, quantile function and random generation for the Poisson-exponential compound probability distribution with parameters fraction and rate.

Usage

```
dpoix(x, frac, rate, log=FALSE)
ppoix(q, frac, rate, lower.tail=TRUE, log.p=FALSE)
qpoix(p, frac, rate, lower.tail=TRUE, log.p=FALSE)
rpoix(n, frac, rate)
```

Arguments

x	vector of (non-negative integer) quantiles. In the context of species abundance distributions, this is a vector of abundances of species in a sample.
q	vector of (non-negative integer) quantiles. In the context of species abundance distributions, a vector of abundances of species in a sample.
n	number of random values to return.
p	vector of probabilities.
frac	single numeric 0 < frac <= 1; fraction of the population or community sampled (see details).
rate	vector of (non-negative) rates of the exponential distribution of the sampled population (see details).
log, log.p	logical; if TRUE, probabilities p are given as log(p)
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].

Details

A compound Poisson-exponential distribution is a Poisson probability distribution where its single parameter lambda, is frac*n, at which n is a random variable with exponential distribution. Thus, the expected value and variance are $E[X] = Var[X] = frac*n$. The density function is

$p(y) = rate*frac^y / (frac + rate)^{(y+1)}$

for x = 0, 1, 2, ... (Green & Plotkin 2007) In ecology, this distribution gives the probability that a species has an abundance of x individuals in a random sample of a fraction frac of the community. In the community the species abundances are independent random variables that follow an exponential density function.

Hence, a Poisson-exponential distribution is a model for species abundances distributions (SAD) in a sample taken from a community under the assumptions: (a) species abundances in the community are independent identically distributed exponential variables, (b) sampling is a Poisson process with expected value 'frac*n', (c) individuals are sampled with replacement, or the fraction of total individuals sampled is small enough to approximate a sample with replacement. See Engen (1977) and Alonso et al. (2008) for critic evaluations.

Notice that the Poisson-exponential can be seen as a different form for the MacArthur's Broken stick model (Baczkowski, 2000), so instead of fitting to a Poisson-exponential distribution directly, the user should use fitbs.

Value

(log) density of the (zero-truncated) density.

Author(s)

Paulo I Prado <prado@ib.usp.br>, Cristiano Strieder and Andre Chalom.

References

Alonso, D. and Ostling, A., and Etienne, R.S. 2008. The implicit assumption of symmetry and the species abundance distribution. *Ecology Letters, 11*: 93–105.

Engen, S. 1977. Comments on two different approaches to the analysis of species frequency data. *Biometrics, 33*: 205–213.

Pielou, E.C. 1977. *Mathematical Ecology*. New York: John Wiley and Sons.

Green,J. and Plotkin, J.B. 2007 A statistical theory for sampling species abundances. *Ecology Letters 10*:1037–1045

See Also

dexp, dpois for related distributions, dpoig for the general case of the Poisson-Gamma distribution

---

dpowbend	*Puyeo's Power-bend discrete distribution*

Description

Density, distribution function, quantile function and random generation for discrete version of Pueyo's power-bend distribution with parameters s and omega.

Usage

```
dpowbend( x, s, omega = 0.01, oM = -log(omega), log=FALSE)
ppowbend ( q, s, omega = 0.01, oM = -log(omega), lower.tail=TRUE, log.p=FALSE)
qpowbend( p, s, omega = 0.01, oM = -log(omega), lower.tail= TRUE, log.p=FALSE)
rpowbend( n, s, omega)
```

Arguments

x	vector of (integer x>0) quantiles. In the context of species abundance distributions, this is a vector of abundances of species in a sample.
q	vector of (integer x>0) quantiles. In the context of species abundance distributions, a vector of abundances of species in a sample.
p	vector of probabilities.
n	number of random values to return.
s	positive real s > 1; exponent of the power-bend distribution.
omega	positive real greater than 0; bending parameter of the distribution. The current implementation only accepts omega smaller than 5.
oM	real number; alternative parametrization which is used for faster fitting on the [fitpowbend](#) function. The current implementation only accepts oM greater than approximately -1.5. You can specify either 'omega' or 'oM', but not both.
log, log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].

Details

The power-bend density is a discrete probability distribution based on the power probability density with an added term for "bending", defined for integer x > 0:

$$p(x) = \frac{x^{-s}e^{-\omega x}}{Li_s(e^{\omega})}$$

The bending term can be seen as a finite-size correction of the power law (Pueyo 2006). Therefore, the power-bend can describe better samples taken from power-law distributions.

The function $Li_s(e^{\omega})$ is the Polylogarithm of the exponential of omega on base s, and represents the integration constant. A naive implementation of the Polylogarithm function is included in the package, and accepts values for non-integer s and omega smaller than 5, which cover the cases of biological relevance.

This distribution was proposed by S. Pueyo to describe species abundance distributions (sads) as a generalization of the logseries distribution. Fisher's logseries correspond to the power-bend with parameters s=1 and $\omega = -\log(\frac{N}{N+\alpha})$ where N is sample size or total number of individuals and $alpha$ is the Fisher's alpha parameter of the logseries.

When fitted to sads, power-law distributions usually overestimates the abundance of common species, and in practice power-bend corrects this problem and usually provides a better fit to abundance distributions.

Value

dpowbend gives the (log) density of the function, ppowbend gives the (log) distribution function, qpowbend gives the quantile function.

Invalid values for parameter s or omega will result in return values NaN, with a warning. Note that integer values of s and omega values larger than 5 are currently not supported, and will also return NaN.

Author(s)

Paulo I Prado <prado@ib.usp.br> and Andre Chalom.

References

Pueyo, S. (2006) *Diversity: Between neutrality and structure*, Oikos 112: 392-405.

See Also

dpower for the power distribution; [dls](dls) for the logseries distribution, which is a particular case of the power-bend, fitpowbend for maximum likelihood estimation in the context of species abundance distributions.

Examples

```
x <- 1:20
PDF <- dpowbend(x=x, s=2.1, omega=0.5)
CDF <- ppowbend(q=x, s=2.1, omega=0.5)
par(mfrow=c(1,2))
plot(x,CDF, ylab="Cumulative Probability", type="b",
     main="Powerbend distribution, CDF")
plot(x,PDF, ylab="Probability", type="h",
     main="Powerbend distribution, PDF")
par(mfrow=c(1,1))

## The powbend distribution is a discrete PDF, hence:
all.equal( ppowbend(10, s=2.1, omega=0.5), sum(dpowbend(1:10, s=2.1, omega=0.5)) ) # should be TRUE

## quantile is the inverse of CDF
all.equal(qpowbend(CDF, s=2.1, omega=0.5), x)

## Equivalence between power-bend and logseries
x <- 1:100
N <- 1000
alpha <- 5
X <- N/(N+alpha)
omega <- -log(X)
PDF1 <- dls(x, N, alpha)
PDF2 <- dpowbend(x, s=1, omega)
plot(PDF1,PDF2, log="xy")
abline(0,1, col="blue")
```

---

dpower   *Power discrete distribution*

Description

Density, distribution function, quantile function and random generation for discrete version of power distribution with parameter s.

Usage

```
dpower( x, s, log=FALSE)
ppower( q, s, lower.tail=TRUE, log.p=FALSE)
qpower( p, s, lower.tail= TRUE, log.p=FALSE)
rpower( n, s, bissection = FALSE, ...)
```

Arguments

x   vector of (integer x>0) quantiles. In the context of species abundance distributions, this is a vector of abundances of species in a sample.

q   vector of (integer x>0) quantiles. In the context of species abundance distributions, a vector of abundances of species in a sample.

n	number of random values to return.
p	vector of probabilities.
s	positive real s > 1; exponent of the power distribution.
log, log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].
bissection	logical; uses the bissection method to generate random numbers? If FALSE calls poweRlaw::rpldis, which provides a faster, continuous approximation for large numbers (Gilliespie 2015).
...	further arguments to be passed to poweRlaw::rpldis when bissection = TRUE

Details

The power density is a discrete probability distribution defined for integer x > 0:

$$p(x) = \frac{x^{-s}}{\zeta(s)}$$

Hence p(x) is proportional to a negative power of 'x', given by the 's' exponent. The Riemann's ζ function is the integration constant.

The power distribution can be used as a species abundance distribution (sad) model, which describes the probability of the abundance 'x' of a given species in a sample or assemblage of species.

Value

dpower gives the (log) density of the density, ppower gives the (log) distribution function, qpower gives the quantile function.

Invalid values for parameter s will result in return values NaN, with a warning.

Author(s)

Paulo I Prado <prado@ib.usp.br> and Murilo Dantas Miranda.

References

Johnson N. L., Kemp, A. W. and Kotz S. (2005) *Univariate Discrete Distributions*, 3rd edition, Hoboken, New Jersey: Wiley. Section 11.2.20.

Colin S. Gillespie (2015) Fitting Heavy Tailed Distributions: The poweRlaw Package. *Journal of Statistical Software, 64(2)*, 1-16. URL http://www.jstatsoft.org/v64/i02/.

See Also

dzeta in **VGAM** package; **poweRlaw** package; [fitpower](#) for maximum likelihood estimation in the context of species abundance distributions.

Examples

```
x <- 1:20
PDF <- dpower(x=x, s=2)
CDF <- ppower(q=x, s=2)
par(mfrow=c(1,2))
plot(x,CDF, ylab="Cumulative Probability", type="b",
     main="Power distribution, CDF")
plot(x,PDF, ylab="Probability", type="h",
     main="Power distribution, PDF")
par(mfrow=c(1,1))

## The power distribution is a discrete PDF, hence:
all.equal( ppower(10, s=2), sum(dpower(1:10, s=2)) ) # should be TRUE

## quantile is the inverse of CDF
all.equal(qpower(CDF, s=2), x)
```

---

dtrunc   *Left-truncation of density, probability and quantile of distributions*

Description

Returns density, probability, quantile values and random generation for distribution functions left-truncated at a specified value.

Usage

```
dtrunc(f, x, trunc, coef, log = FALSE)
ptrunc(f, q, trunc, coef, lower.tail=TRUE, log.p=FALSE)
qtrunc(f, p, trunc, coef, lower.tail = TRUE, log.p = FALSE)
rtrunc(f, n, trunc, coef, ...)
```

Arguments

f	character; root name of the density or distribution function to be truncated - e.g., "lnorm" for the lognormal distribution; "geom" for the geometric distribution.
x, q	vector of quantiles.
trunc	numeric, trunc > min(x). Truncation value (see details).
p	vector of probabilities.
n	number of random values to return.
coef	numeric named list; parameters values of the density or distribution function, named accordingly (see details).
log, log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].
...	in rtrunc further arguments to be passed to poweRlaw::rpldis when argument argument f = "power" and argument bissection = FALSE. The same for rpower when and argument bissection = FALSE.

Details

Given a distribution with probability distribution function (PDF) g and cumulative distribution function (CDF) G, a random variable x with these distributions left-truncated at `trunc` has its PDF:

g'(x) = g(x)/(1 - G(trunc)) for any x <= trunc and zero otherwise

and CDF:

G'(x) = (G(max(x,trunc)) - G(trunc)) / (1 - G(trunc))

`dtrunc` and `ptrunc` calculates the left-truncated distributions functions g'(x) and G'(x) defined above for a vector of values x from any standard distribution function available in R. This means the 'upper tail' of a continuous distribution is rescaled to integrate to one. Accordingly, for discrete distributions, the probabilities for all x>trunc are rescaled to sum one. `qtrunc` is the inverse function of `ptrunc`.

Left-truncated distributions can be used to describe the species abundance distributions (SADs), specially for continuous distributions (e.g., truncated lognormal distribution).

Value

`dtrunc` gives the (log) density defined by f left-truncated at `trunc`. `ptrunc` gives the (log) distribution function defined by f left-truncated at `trunc`. `qtrunc` gives the quantile of the density defined by f left-truncated at `trunc`. `rtrunc` generates a sample from the density defined by f left-truncated at `trunc`.

Source

Codes from Nadarajah and Kotz (2006), which provide a more generic solution for left and right truncation.

References

Nadarajah, S. and Kotz, S. 2006. R Programs for Computing Truncated Distributions. *Journal of Statistical Software 16*:Code Snippet 2.

See Also

Distributions for standard distributions in R; many functions in package **sads** have an argument `trunc` that allows to simulate and fit truncated distributions for species abundance distributions (e.g., [fitsad rsad](), [radpred](), [octavpred](). Package 'VGAM' has truncated versions of many standard functions; see `Truncate-methods` in package **distr** for general methods to build R objects of truncated distributions.

Examples

```
A <- dtrunc("lnorm", x = 1:5, trunc = 0.5,
       coef = list( meanlog=1, sdlog=1.5 ) )
## same as
B <- dlnorm( 1:5 , meanlog = 1, sdlog = 1.5 ) /
  ( plnorm ( 0.5 , meanlog = 1, sdlog = 1.5, lower = FALSE))
## checking
identical( A, B )
```

```
A <- ptrunc("pois", q = 1:5, trunc = 0,
       coef = list( lambda = 1.5 ) )
## same as
B <- (ppois( 1:5 , lambda = 1.5 ) -
       ppois(0 , lambda = 1.5 ) ) /
  (ppois(0 , lambda = 1.5, lower = FALSE))
## checking
identical(A,B)

# Random generation
rtrunc("ls", 100, coef=list(N=1000, alpha=50), trunc=5)
```

---

dvolkov   *Neutral Biodiversity Theory distribution by Volkov* et al.

Description

Density, distribution function, quantile function and random generation for species abundances
distribution in a neutral community with immigration as deduced by Volkov et al. (2003).

Usage

```
dvolkov( x, theta, m, J, order=96, log = FALSE )
pvolkov( q, theta, m , J, lower.tail = TRUE, log.p = FALSE )
qvolkov( p, theta, m, J, lower.tail = TRUE, log.p = FALSE )
rvolkov( n, theta, m, J)
Svolkov( theta, m, J, order=96)
```

Arguments

x	vector of (non-negative integer) quantiles. In the context of species abundance distributions, this is a vector of abundance of species in a sample.
q	vector of (non-negative integer) quantiles. In the context of species abundance distributions, a vector of abundance of species in a sample.
n	number of random values to return.
p	vector of probabilities.
theta	positive real, theta > 0; Hubbell's 'fundamental biodiversity number'.
order	order of the approximation for the numerical integration. The default of 96 usually gives a rounding error of 1e-8 for moderate dataset; orders greater than 128 are probably overkill
m	positive real, 0 <= m <= 1; immigration rate (see details).
J	positive integer; sample size. In the context of species abundance distributions, usually the number of individuals in a sample.
log, log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].

Details

Volkov *et al* (2003) proposed one of the analytic solutions for the species abundance distributions (SADs) for The Neutral Theory of Biodiversity (Hubbell 2001).

Their solution is deduced from a model of stochastic dynamics of a set of species where the following rules apply: (1) replacement of a dead individual by local offspring – with probability 1-m, individuals picked at random are replaced by the offspring of other individuals picked at random; (2) replacement of a dead individual by an immigrant – with probability m individuals picked at random are replaced by immigrants taken at random from a pool of potential colonizers (the metacommunity).

Volkov et al. (2003, eq.7) provide the stationary solution for the expected number of species with a given abundance. A probability density function is easily calculated by taking these expected values for abundances 1:J and dividing them by the total number of species. dvolkov performs the numerical integration of the density function by means of Gaussian quadrature, using a library by Pavel Holoborodko (http://www.holoborodko.com/pavel/?page_id=679). The code is based on the untb::volkov function (Hankin 2007). pvolkov provides CDF by cumulative sum of density values, and qvolkov use a numeric interpolation with a step function (approxfun) to find quantiles. Calculations can be slow for larger datasets.

A special function Svolkov is provided to estimate the expected community size from a Volkov distribution with parameters theta, m and J, but this function is deprecated. A more comprehensive function to estimate community sizes will be developed in the future.

Value

dvolkov gives the (log) density of the density, pvolkov gives the (log) distribution function, qvolkov gives the (log) the quantile function.

Invalid values for parameters J or theta will result in return values NaN, with a warning.

Author(s)

Paulo I Prado <prado@ib.usp.br>, Andre Chalom and Murilo Dantas Miranda.

References

Hankin, R.K.S. 2007. Introducing untb, an R Package For Simulating Ecological Drift Under the Unified Neutral Theory of Biodiversity. *Journal of Statistical Software 22* (12).

Hubbell, S. P. 2001. *The Unified Neutral Theory of Biodiversity*. Princeton University Press.

Volkov, I., Banavar, J.R., Hubbell, S.P., Maritan, A. 2003. Neutral theory and relative species abundance in ecology. *Nature 424*:1035–1037

See Also

fitvolkov for maximum likelihood fit, dmzsm for the distribution of abundances in the metacommunity, volkov in package **untb**.

Examples

```
## Volkov et al 2003 fig 1
## But without Preston correction to binning method
## and only the line of expected values by Volkov's model
data( bci )
bci.oct <- octav( bci, preston = FALSE )
plot( bci.oct )
CDF <- pvolkov( bci.oct$upper, theta = 47.226, m = 0.1, J = sum(bci) )
bci.exp <- diff( c(0,CDF) ) * length(bci)
midpoints <- as.numeric( as.character( bci.oct$octave ) ) - 0.5
lines( midpoints, bci.exp, type="b" )

## the same with Preston binning using octavpred
plot(octav( bci, preston = TRUE ))
bci.exp2 <- octavpred( bci, sad = "volkov",
                       coef = list(theta = 47.226, m = 0.1, J=sum(bci)), preston=TRUE)
lines( bci.exp2 )
```

---

   dzipf   *Zipf distribution*

Description

Density, distribution function, quantile function and random generation for Zipf distribution with parameters N and s.

Usage

```
dzipf( x, N, s, log=FALSE)
pzipf( q, N, s, lower.tail=TRUE, log.p=FALSE)
qzipf( p, N, s, lower.tail = TRUE, log.p = FALSE)
rzipf( n, N, s)
```

Arguments

x	vector of (non-negative integer) quantiles. In the context of species abundance distributions, this is a vector of abundance ranks of species in a sample.
q	vector of (non-negative integer) quantiles. In the context of species abundance distributions, a vector of abundance ranks of species in a sample.
n	number of random values to return.
p	vector of probabilities.
N	positive integer $0 < N <$ Inf, total number of elements of a collection. In the context of species abundance distributions, usually the number of species in a sample.
s	positive real $s > 0$; Zipf's exponent
log, log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].

Details

The Zipf distribution describes the probability or frequency of occurrence of a given element from a set of N elements. According to Zipf's law, this probability is inversely proportional to a power s of the frequency rank of the element in the set. The density function is

$$p(x) = \frac{x^{-s}}{\sum_{i=1}^{N} i^{-s}}$$

Since p(x) is proportional to a power of x, the Zipf distribution is a power distribution. The Zeta distribution is a special case at the limit N -> Inf.

The Zipf distribution has a wide range of applications (Li 2011). One of its best known applications is describing the probability of occurrence of a given word that has a ranking x in a *corpus* with a total of N words. It can also be used to describe the probability of the abundance rank of a given species in a sample or assemblage of N species.

Value

dzipf gives the (log) density, pzipf gives the (log) distribution function, qzipf gives the quantile function.

Author(s)

Paulo I Prado <prado@ib.usp.br> and Murilo Dantas Miranda.

References

Johnson N. L., Kemp, A. W. and Kotz S. (2005) *Univariate Discrete Distributions*, 3rd edition, Hoboken, New Jersey: Wiley. Section 11.2.20.

Li, W. (2002) Zipf's Law everywhere. *Glottometrics* 5:14-21

Zipf's Law. https://en.wikipedia.org/wiki/Zipf's_law.

See Also

dzipf and rzipf and related functions in **zipfR** package; Zeta for zeta distribution in **VGAM** package. fitzipf to fit Zipf distribution as a rank-abundance model.

Examples

```
x <- 1:20
PDF <- dzipf(x=x, N=100, s=2)
CDF <- pzipf(q=x, N=100, s=2)
par(mfrow=c(1,2))
plot(x,CDF, ylab="Cumulative Probability", type="b",
     main="Zipf distribution, CDF")
plot(x,PDF, ylab="Probability", type="h",
     main="Zipf distribution, PDF")
par(mfrow=c(1,1))

## quantile is the inverse of CDF
```

```
all.equal( qzipf(CDF, N=100, s=2), x) # should be TRUE

## Zipf distribution is discrete hence
all.equal( sum(dzipf(1:10, N=10, s=2)), pzipf(10, N=10, s=2)) # should be TRUE
```

fitrad	*ML fitting of species rank-abundance distributions*

Description

Fits probability distributions for abundance ranks of species in a sample or assemblage by maximum likelihood.

Usage

```
fitrad(x, rad =c("gs", "mand", "rbs", "zipf"), ...)
fitgs(x, trunc, start.value, ...)
fitmand(x, trunc, start.value, ...)
fitrbs(x, trunc, ...)
fitzipf(x, N, trunc, start.value, upper = 20, ...)
```

Arguments

x	vector of (positive integer) quantiles or an object of [rad-class](). In the context of rads, the numerical vector contains abundances of species in a sample or ecological assemblage according to their abundance. The [rad-class]() object contains ranked abundances of species in a sample or ecological assemblage.
rad	character; root name of community rad distribution to be fitted. "gs" for geometric series (not geometric distribution, [dgeom]()), "mand" for Zipf-Mandelbrodt distribution, "rbs" for MacArthur's Broken-stick distribution, "zipf" for Zipf distribution.
trunc	non-negative integer, trunc > min(x); truncation point to fit a truncated distribution.
N	positive integer, total number of individuals in the sample/assemblage.
start.value	numeric named vector; starting values of free parameters to be passed to mle2. Parameters should be named as in the corresponding density function, and in the same order.
upper	real positive; upper bound for the Brent's one-parameter optimization method (default), for fits that use this method by default. See details and optim.
...	in fitrad further arguments to be passed to the specific fitting function (most used are trunc and start.value). In the specific fitting functions further arguments to be passed to mle2.

Details

All these functions fit rank-abundance distributions (RAD) to a vector of abundances or a rank-abundance table of the rad-class. RADs assign probabilities p(i) to each rank i, which can be interpreted as the expected proportion of total individuals in the sample that are of the i-th species.

fitrad is simply a wrapper that calls the specific functions to fit the distribution chosen with the argument rad. Users can interchangeably use fitrad or the individual functions detailed below (e.g. fitrad(x, sad="rbs", ...) is the same as fitrbs(x, ...) and so on).

The distributions are fitted by the maximum likelihood method using numerical optimization, with mle2. The resulting object is of fitrad-class which can be handled with mle2 methods for fitted models and has also some additional methods for RADs models (see fitrad-class and examples). By default, fitting to one-parameter distributions (fitgs, fitzipf) uses Brent's one-dimensional method of optimization (see optim).

fitgs fits Motomura's Geometric Series (Whittaker 1965, May 1975) to abundance ranks. This was the first model fitted to species abundance data (Motomura 1932, *apud* Doi and Mori 2012), which was subsequently described as the result of niche pre-emption at a constant rate (Numata et. al. 1953 *apud* Doi and Mori 2012). The initial guess for parameter k is given by the expression 1 - (nmin/nmax)^(1/(S-1)) (He & Tang, 2008).

fitrbs fits the Broken-stick distribution (MacArthur 1960) to abundance ranks. It is defined only by the observed number of elements S in the collection and collection size N. Therefore, once a sample is taken, the Broken-stick has no free parameters. Therefore, there is no actual fitting, but still the fitrbs calls mle2 with fixed parameters N and S and eval.only=TRUE to return an object of fitrad-class to keep compatibility with other RAD models fitted to the same data. Therefore the resulting objects allows most of the operations with RAD models, such as comparison with other models through model selection, diagnostic plots and so on (see fitrad-class).

fitzipf and fitmand fit the Zipf distribution and its two-parameter generalization, the Zipf-Mandelbrodt distribution. Both are discrete power-law distributions commonly proposed as RAD models, though they in general provide poor fit to species abundances (Newman 2005).

Value

An object of fitrad-class which inherits from mle2-class and thus has methods for handling results of maximum likelihood fits from mle2 and also specific methods to handle rank-abundance models.

Author(s)

Paulo I Prado <prado@ib.usp.br>, Andre Chalom and Murilo Dantas Miranda

Source

all fitting functions builds on mle2 and methods from 'bbmle' package (Bolker 2012), which in turn builds on mle function and associated classes and methods.

References

Bolker, B. and R Development Core Team 2012. bbmle: Tools for general maximum likelihood estimation. R package version 1.0.5.2. http://CRAN.R-project.org/package=bbmle

Doi, H. and Mori, T. 2012. The discovery of species-abundance distribution in an ecological community. *Oikos 122:* 179–182.

He, F. and Tang, D. 2008. Estimating the niche preemption parameter of the geometric series. *Acta Oecologica 33:* 105–107.

MacArthur, R.H. 1960. On the relative abundance of species. *Am Nat 94*:25–36.

May, R.M. 1975. Patterns of Species Abundance and Diversity. In Cody, M.L. and Diamond, J.M. (Eds) *Ecology and Evolution of Communities*. Harvard University Press. pp 81–120.

Newman, M.E.J. 2005. Power laws, Pareto distributions and Zipf's law. *Contemporary Physics, 46:* 323–351.

Whittaker, R.H. 1965. Dominance and diversity in land plant communities. *Science 147*: 250–260.

See Also

dgs, dmand, drbs, dzipf, for corresponding density functions created for fitting RADs; fitrad-class.

Examples

```
## Figure 2 of Motomura (1932)
data(okland)
plot(rad(okland))
ok.gs <- fitrad(okland, "gs")
lines(radpred(ok.gs))

## Comparison with Zipf-Mandelbrodt
ok.zm <- fitrad(okland, "mand")
AICctab(ok.gs, ok.zm, nobs=length(okland))
lines(radpred(ok.zm), col="red")
```

---

fitrad-class	*Class* "fitrad" *for maximum likelihood fitting of species rank-abundance distributions*

Description

This class extends mle2-class to encapsulate models of species rank-abundance distributions (RADs) fitted by maximum likelihood.

Objects from the Class

Objects created by a call to function fitrad, which fits a probability distribution to an abundance vector.

Slots

rad: Object of class "character"; root name of the species abundance distribution fitted. See man page of fitrad for available models.

distr: Deprecated since sads 0.2.4. See distr function

trunc: Object of class "numeric"; truncation value used in the fitted model. 'NA' for a non-truncated distribution.

rad.tab: Object of class "rad"; rank-abundance table of observed abundances.

call: Object of class "language"; The call to [mle2](mle2).

call.orig: Object of class "language" The call to [mle2](mle2), saved in its original form (i.e. without data arguments evaluated).

coef: Object of class "numeric"; Vector of estimated parameters.

fullcoef: Object of class "numeric"; Fixed and estimated parameters.

vcov: Object of class "matrix"; Approximate variance-covariance matrix, based on the second derivative matrix at the MLE.

min: Object of class "numeric"; Minimum value of objective function = minimum negative log-likelihood.

details: Object of class "list"; Return value from optim.

minuslogl: Object of class "function"; The negative log-likelihood function.

method: Object of class "character"; The optimization method used.

data: Object of class "data.frame"; Data with which to evaluate the negative log-likelihood function.

formula: Object of class "character"; If a formula was specified, a character vector giving the formula and parameter specifications.

optimizer: Object of class "character"; The optimizing function used.

Extends

Class "[mle2](mle2)", directly.

Methods

octavpred signature(object = "fitrad", sad = "missing",rad = "missing", coef = "missing", trunc = "missing", oct = "ANY", S = "missing", N = "missing"): expected number of species per abundance octave, see octav and octavpred.

plot signature(x = "fitrad", y = "ANY"): diagnostic plots of the fitted model.

show signature(object = "fitrad"): Displays object.

nobs signature(object = "fitrad"): Displays number of observations (number of species) in the data to which the model was fitted.

pprad signature(x = "fitrad", sad = "missing", coef = "missing", trunc = "missing"): plot of observed vs predicted percentiles of the abundance distribution, details in pprad.

qqrad signature(x = "fitrad", sad = "missing", coef = "missing", trunc = "missing"): plot of observed vs predicted quantiles of the abundance distribution, details in qqrad.

radpred signature(object = "fitrad", sad = "missing",rad = "missing", coef = "missing", trunc = "missing", distr = "missing", S = "missing", N = "missing"): expected abundances of the 1st to n-th most abundant species, see [rad](rad) and radpred.

Note

Class `fitrad` only adds four slots to class `mle2`. The descriptions of slots inherited from `mle2-class` replicate those in `mle2-class`.

Author(s)

Paulo I Prado <prado@ib.usp.br> and Murilo Dantas Miranda, after Ben Bolker and R Core Team.

Source

this class builds on `mle2-class` of **bbmle** package (Bolker 2012), which in turn builds on `mle-class`.

References

Bolker, B. and R Development Core Team 2012. bbmle: Tools for general maximum likelihood estimation. R package version 1.0.5.2. http://CRAN.R-project.org/package=bbmle

See Also

`mle2-class` for all methods available from which `fitrad-class` inherits; `fitrad` for details on fitting RADs models; `octavpred` and `radpred` to get rank-abundance and frequencies of species in octaves predicted from fitted models.

Examples

```
ok.gser <- fitrad(okland, "gs")
## The class has a plot method to show diagnostic plots
par(mfrow=c(2,2))
plot(ok.gser)
# The same plot, but with relative abundances
plot(ok.gser, prop = TRUE)
par(mfrow=c(1,1))
## Some useful methods inherited from mle2-class
coef(ok.gser)
confint(ok.gser)
logLik(ok.gser)
## Model selection
ok.zipf <- fitrad(okland, "zipf")
AICctab(ok.gser, ok.zipf, nobs=length(moths), base=TRUE)
```

---

fitsad   *ML fitting of species abundance distributions*

Description

Fits probability distributions for abundances of species in a sample or assemblage by maximum likelihood.

Usage

```
fitsad(x, sad = c("bs","exp", "gamma","geom","lnorm","ls","mzsm","nbinom","pareto",
       "poilog","power", "powbend", "volkov","weibull"), ...)

fitbs(x, trunc, ...)

fitexp(x, trunc = NULL, start.value, ...)

fitgamma(x, trunc, start.value,  ...)

fitgeom(x, trunc = 0, start.value, ...)

fitlnorm(x, trunc, start.value,  ...)

fitls(x, trunc, start.value, upper = length(x), ...)

fitmzsm(x, trunc, start.value, upper = length(x), ...)

fitnbinom(x, trunc = 0, start.value, ...)

fitpareto(x, trunc, start.value,  upper = 20, ...)

fitpoilog(x, trunc = 0, ...)

fitpower(x, trunc, start.value, upper = 20, ...)

fitpowbend(x, trunc, start.value, ...)

fitvolkov(x, trunc, start.value, ...)

fitweibull(x, trunc, start.value,  ...)
```

Arguments

x	vector of (positive integer) quantiles. In the context of SADs, some abundance measurement (e.g., number of individuals, biomass) of species in a sample or ecological assemblage.
sad	character; root name of community sad distribution to be fitted. "bs" for broken-stick distribution, "exp" for exponential distribution, "gamma" for gamma distribution, "geom" for geometric distributions (not geometric series rad model, [dgs](#)), "lnorm" for lognormal, "ls" for Fisher's log-series, "mzsm" for Alonso & McKane's neutral metacommunity distribution, "nbinom" for negative binomial, "pareto" for Pareto distribution, "poilog" for Poisson-lognormal distribution, "power" for power-law distribution, "powbend" for Pueyo's power-bend distribution, "volkov" for Volkov's et al. neutral community distribution, "weibull" for Weibull distribution.
trunc	non-negative integer, trunc > min(x); truncation point to fit a truncated distribution. For "poilog", "geom" and "nbinom" set trunc=NULL to avoid zero-

	truncation (see details).
start.value	named numeric vector; starting values of free parameters to be passed to mle2. Parameters should be named as in the corresponding density function, and in the same order.
upper	real positive; upper bound for Brent's one-parameter optimization method (default), for fits that use this method by default. See details and optim.
...	in fitsad further arguments to be passed to the specific fitting function (most used are trunc, start.value) In the specific fitting functions further arguments to be passed to mle2.

Details

fitsad is simply a wrapper that calls the specific functions to fit the distribution chosen with the argument sad. Users can interchangeably use fitsad or the individual functions detailed below (e.g. fitsad(x, sad="geom", ...) is the same as fitgeom(x, ...) and so on).

The distributions are fitted by the maximum likelihood method using numerical optimization, with mle2. The resulting object is of fitsad-class which can be handled with mle2 methods for fitted models and has also some additional methods for SADs models (see fitsad-class and examples).

Functions fitexp, fitgamma, fitlnorm and fitweibull fit the standard continuous distributions most used as SADs models. As species with null abundances in the sample are in general unknown, the fit to continuous distributions can be improved by truncation to some value above zero (see example). Convergence problems can occur when fitting with truncation, and can be solved with sensible starting values. fitgamma uses Chapman's (1956) fitting method to find starting values for the truncated gamma distribution, and fitweibull uses Rinne's (2009, p. 413) method (thanks to Mario Jose Marques-Azevedo).

Functions fitgeom, fitnbinom fits geometric and negative binomial distributions which are two discrete standard distributions also used to fit SADs. Since species with zero individuals in the sample are in general unknown, these functions fit by default zero-truncated distributions. To avoid zero-truncation set trunc=NULL. Using the geometric distribution as a SAD model is not to be confounded for fitting the Geometric series fitgs as a rank-abundance distribution (RAD) model.

Function fitls implements the original numerical recipe by Fisher (1943) to fit the log-series distribution, given a vector of species abundances. Alonso et al. (2008, supplementary material) showed that this recipe gives the maximum likelihood estimate of Fisher's alpha, the single parameter of the log-series. Fitting is done through numerical optimization with the uniroot function, following the code of the function fishers.alpha of the **untb** package. After that, the estimated value of alpha parameter is used as the starting value to get the Log-likelihood from the log-series density function dls, using the function mle2. The total number of individuals in the sample, N, is treated as a fixed parameter in this implementation, in order to maintain coherence with similar parameters from fitvolkov and fitmzsm (see below). Fixed parameters in the model specification do not contribute to the model degrees of freedom, and are not accounted in standard error calculations.

Functions fitpower, fitpowbend and fitpareto fit power-law distributions with one and two-parameters, that have been suggested as SADs models. The implementations of power and power-bend are for discrete distributions that do not include zeroes. The Pareto distribution is continuous and includes all non-negative numbers. Fisher's logseries are a special case of the power-bend, see dpowbend and Pueyo (2006).

Function `fitbs` fits the Broken-stick distribution (MacArthur 1960). It is defined only by the ob-served number of elements S in the collection and collection size N. Thus once a sample is taken, the Broken-stick has no free parameters. Therefore, there is no actual fitting, but still `fitbs` calls `mle2` with fixed parameters N and S and eval.only=TRUE to return an object of class fitsad to keep compatibility with other SADs models fitted to the same data. Therefore, the resulting objects allows most of the operations with SAD models, such as comparison with other models through model selection, diagnostic plots and so on (see `fitsad-class`).

Function `fitpoilog` fits the Poisson-lognormal distribution. This is a compound distributions that describes the abundances of species in Poisson sample of community that follows a lognormal SAD. This is a sampling model of SAD, which is approximated by the 'veil line' truncation of the lognormal (Preston 1948). The Poisson-lognormal is the analytic solution for this sampling model, as Fisher's log-series is a analytic limit case for a Poisson-gamma (a.k.a negative binomial) distribution. As geometric and negative binomial distributions, the Poisson-lognormal includes zero, but the fit is zero-truncated by default, as for `fitgeom`, `fitnbinom`. To avoid zero-truncation set trunc=NULL.

`fitmzsm` fits the metacommunity Zero-sum multinomial distribution `dmzsm` from the Neutral Theory of Biodiversity (Alonso and McKane 2004). The mZSM describes the SAD of a sample taken from a neutral metacommunity under random drift. It has two parameters, the number of individuals in the sample J and `theta`, the 'fundamental biodiversity number'. Because J is known from the sample size, the fit resumes to estimate a single parameter, theta. By default, `fitmzsm` fits mZSM to a vector of abundances with Brent's one-dimensional method of optimization (see `optim`). The log-series distribution (Fisher et al. 1943) is a limiting case of mZSM (Hubbel 2001), and theta tends to Fisher's alpha as J increases. In practice the two models provide very similar fits to SADs (see example).

Function `fitvolkov` fits the SAD model for a community under neutral drift with immigration (Volkov et al. 2003). The model is a stationary distribution deduced from a stochastic process compatible with the Neutral Theory of Biodiversity (Hubbell 2001). It has two free parameters, the 'fundamental biodiversity number' `theta`, and the immigration rate m (see `dvolkov`) fitvolkov builds on function volkov from package **untb** to fit Volkov's *et al.* SAD model to a vector of abundances. The fit can be extremely slow even for vectors of moderate size.

Value

An object of `fitsad-class` which inherits from `mle2-class` and thus has methods for handling results of maximum likelihood fits from `mle2` and also specific methods to handle SADs models (see `fitsad-class`).

Author(s)

Paulo I Prado <prado@ib.usp.br>, Murilo Dantas Miranda and Andre Chalom, after Ben Bolker, R Core Team, Robin Hanking, Vidar Grøtan and Steinar Engen.

Source

all fitting functions builds on `mle2` and methods from **bbmle** package (Bolker 2012), which in turn builds on `mle` function and associated classes and methods; `fitls` and `fitvolkov` use codes and functions from **untb** package (Hankin 2007); `fitpoilog` builds on **poilog** package (Grøtan & Engen 2008).

References

Alonso, D. and McKane, A.J. 2004. Sampling Hubbell's neutral model of biodiversity. *Ecology Letters 7*:901–910

Alonso, D. and Ostling, A., and Etienne, R.S. 2008 The implicit assumption of symmetry and the species abundance distribution. *Ecology Letters, 11*: 93-105.

Bolker, B. and R Development Core Team 2012. bbmle: Tools for general maximum likelihood estimation. R package version 1.0.5.2. http://CRAN.R-project.org/package=bbmle

Chapman, D. G. 1956. Estimating the parameters of a truncated gamma distribution. *The Annals of Mathematical Statistics, 27(2)*: 498–506.

Fisher, R.A, Corbert, A.S. and Williams, C.B. (1943) The Relation between the number of species and the number of individuals in a random sample of an animal population. *The Journal of Animal Ecology, 12(1)*: 42–58.

Grøtan, V. and Engen, S. 2008. poilog: Poisson lognormal and bivariate Poisson lognormal distribution. R package version 0.4.

Hankin, R.K.S. 2007. Introducing untb, an R Package For Simulating Ecological Drift Under the Unified Neutral Theory of Biodiversity. *Journal of Statistical Software 22* (12).

Hubbell, S.P. 2001. *The Unified Neutral Theory of Biodiversity*. Princeton University Press

MacArthur, R.H. 1960. On the relative abundance of species. *Am Nat 94*:25–36.

Magurran, A.E. 1989. *Ecological diversity and its measurement*. Princenton University Press.

Preston, F.W. 1948. The commonness and rarity of species. *Ecology 29*: 254–283.

Pueyo, S. (2006) *Diversity: Between neutrality and structure*, Oikos 112: 392-405.

Rinne, H. 2009. *The Weibull distribution: a handbook*. CRC Press

Volkov, I., Banavar, J. R., Hubbell, S. P., Maritan, A. 2003. Neutral theory and relative species abundance in ecology. *Nature 424*:1035–1037.

See Also

`dls`, `dmzsm`, `dpareto`, `dpoilog`, `dpower`, `dvolkov`, `dpowbend` for corresponding density functions created for fitting SADs; standard distributions dexp, dgamma, dgeom, dlnorm, dnbinom, dweibull; `fitsad-class`.

Examples

```
## Magurran (1989) example 5:
## birds in an Australian forest
mag5 <- c(103, 115, 13, 2, 67, 36, 51, 8, 6, 61, 10, 21,
          7, 65, 4, 49, 92, 37, 16, 6, 23, 9, 2, 6, 5, 4,
          1, 3, 1, 9, 2)
mag5.bs <- fitsad(mag5, "bs")
summary(mag5.bs)## no estimated coefficient
coef(mag5.bs) ## fixed coefficients N and S
## Diagnostic plots
par(mfrow=c(2, 2))
plot(mag5.bs)
par(mfrow=c(1, 1))
```

```
data(moths) #Fisher's moths data
moths.mzsm <- fitmzsm(moths) ## same as fitsad(moths, sad="mzsm")
## fit to log-series
moths.ls <- fitsad(moths, sad="ls")
coef(moths.ls)
coef(moths.mzsm) ## Compare with theta=38.9, Alonso&McKanne (2004)
## Diagnostic plots
par(mfrow=c(2, 2))
plot(moths.mzsm)
par(mfrow=c(1, 1))
## Graphical comparison
plot(rad(moths))
lines(radpred(moths.ls))
lines(radpred(moths.mzsm), col="red", lty=2)
legend("topright", c("log-series", "mZSM"), lty=1, col=c("blue","red"))
## Two more models: truncated lognormal and Poisson-lognormal
moths.ln <- fitsad(moths, "lnorm", trunc=0.5)
moths.pln <- fitsad(moths, "poilog")
## Model selection
AICtab(moths.ln, moths.pln, moths.ls, moths.mzsm, weights=TRUE)

## Biomass as abundance variable
data(ARN82.eB.apr77) #benthonic marine animals
AR.ln <- fitsad(ARN82.eB.apr77, sad="lnorm")
AR.g <- fitsad(ARN82.eB.apr77, sad="gamma")
AR.wb <- fitsad(ARN82.eB.apr77, sad="weibull")
plot(octav(ARN82.eB.apr77))
lines(octavpred(AR.ln))
lines(octavpred(AR.g), col="red")
lines(octavpred(AR.wb), col="green")
legend("topright", c("lognormal", "gamma", "weibull"),lty=1, col=c("blue","red", "green"))
AICctab(AR.ln, AR.g, AR.wb, nobs=length(ARN82.eB.apr77), weights=TRUE)
```

---

fitsad-class	*Class* "fitsad" *for maximum likelihood fitting of species abundance distributions*

Description

This class extends `mle2-class` to encapsulate models of species abundance distributions (SADs) fitted by maximum likelihood.

Objects from the Class

Objects created by a call to function `fitsad`, which fits a probability distribution to an abundance vector.

Slots

sad: Object of class "character"; root name of the species abundance distribution fitted. See man page of fitsad for available models.

distr: Deprecated since sads 0.2.4. See distr function

trunc: Object of class "numeric"; truncation value used in the fitted model. 'NA' for a non-truncated distribution.

call: Object of class "language"; The call to mle2.

call.orig: Object of class "language" The call to mle2, saved in its original form (i.e. without data arguments evaluated).

coef: Object of class "numeric"; Vector of estimated parameters.

fullcoef: Object of class "numeric"; Fixed and estimated parameters.

vcov: Object of class "matrix"; Approximate variance-covariance matrix, based on the second derivative matrix at the MLE.

min: Object of class "numeric"; Minimum value of objective function = minimum negative log-likelihood.

details: Object of class "list"; Return value from optim.

minuslogl: Object of class "function"; The negative log-likelihood function.

method: Object of class "character"; The optimization method used.

data: Object of class "data.frame"; Data with which to evaluate the negative log-likelihood function.

formula: Object of class "character"; If a formula was specified, a character vector giving the formula and parameter specifications.

optimizer: Object of class "character"; The optimizing function used.

Extends

Class "mle2", directly.

Methods

octavpred signature(object = "fitsad", sad = "missing",rad = "missing", coef = "missing", trunc = "missing", oct = "ANY", S = "missing", N = "missing"): expected number of species per abundance octave, see octav and octavpred.

plot signature(x = "fitsad", y = "ANY"): diagnostic plots of the fitted model.

nobs signature(object = "fitsad"): Displays number of observations (number of species) in the data to which the model was fitted.

show signature(object = "fitsad"): Displays object.

ppsad signature(x = "fitsad", sad = "missing", coef = "missing", trunc = "missing"): plot of observed vs predicted percentiles of the abundance distribution, details in ppsad.

qqsad signature(x = "fitsad", sad = "missing", coef = "missing", trunc = "missing", distr = "missing"): plot of observed vs predicted quantiles of the abundance distribution, details in qqsad.

radpred signature(object = "fitsad", sad = "missing",rad = "missing", coef = "missing", trunc = "missing", distr = "missing", S = "missing", N = "missing"): expected abundances of the 1st to n-th most abundant species, see rad and radpred.

Note

Class `fitsad` only adds three slots to class `mle2`. The descriptions of slots inherited from `mle2-class` replicate those in mle2-class.

Author(s)

Paulo I Prado <prado@ib.usp.br> and Murilo Dantas Miranda, after Ben Bolker and R Core Team.

Source

this class builds on `mle2-class` of **bbmle** package (Bolker 2012), which in turn builds on `mle-class`.

References

Bolker, B. and R Development Core Team 2012. bbmle: Tools for general maximum likelihood estimation. R package version 1.0.5.2. http://CRAN.R-project.org/package=bbmle

See Also

mle2-class for all methods available from which fitsad-class inherits; fitsad for details on fitting SADs models; octavpred and radpred to get rank-abundance and frequencies of species in octaves predicted from fitted models.

Examples

```
moths.ls <- fitsad(moths, "ls")
## The class has a plot method to show diagnostic plots
par(mfrow=c(2,2))
plot(moths.ls)
# the same plot, but with relative abundances
plot(moths.ls, prop = TRUE)
par(mfrow=c(1,1))
## Some useful methods inherited from mle2-class
coef(moths.ls)
confint(moths.ls)
logLik(moths.ls)
## Model selection
moths.ln <- fitsad(moths, "lnorm", trunc=0.5)
AICctab(moths.ls, moths.ln, nobs=length(moths), base=TRUE)
```

---

fitsadC	*ML fitting of species abundance distributions to data pooled in abundance classes*

Description

Fits probability distributions for abundances of species aggregated in abundance classes in a sample or assemblage by maximum likelihood.

Usage

```
fitsadC(x, sad = c("exp","gamma","lnorm","pareto", "weibull"), ...)

fitexpC(x, trunc = NULL, start.value, ...)

fitgammaC(x, trunc, start.value,  ...)

fitlnormC(x, trunc, start.value,  ...)

fitparetoC(x, trunc, start.value,  upper = 20, ...)

fitweibullC(x, trunc, start.value,  ...)
```

Arguments

x	object of class `histogram` with the number of species in abundance classes, in a sample or ecological assemblage.
sad	character; root name of community sad distribution to be fitted. `"exp"` for exponential distribution, `"gamma"` for gamma distribution, `"lnorm"` for lognormal, `"pareto"` for Pareto distribution, `"weibull"` for Weibull distribution.
trunc	non-negative integer, `trunc > min(x)`; truncation point to fit a truncated distribution.
start.value	named numeric vector; starting values of free parameters to be passed to `mle2`. Parameters should be named as in the corresponding density function, and in the same order.
upper	real positive; upper bound for Brent's one-parameter optimization method (default), for fits that use this method by default. See details and `optim`.
...	in `fitsad` further arguments to be passed to the specific fitting function (most used are `trunc`, `start.value`) In the specific fitting functions further arguments to be passed to `mle2`.

Details

`fitsadC` is simply a wrapper that calls the specific functions to fit the distribution chosen with the argument `sad`. Users can interchangeably use `fitsadC` or the individual functions detailed below (e.g. `fitsad(x, sad="exp", ...)` is the same as `fitexpC(x, ...)` and so on).

The distributions are fitted by the maximum likelihood method using numerical optimization, with `mle2`. The resulting object is of `fitsadC-class` which can be handled with `mle2` methods for fitted models and has also some additional methods for SADs models (see fitsadC-class and examples).

For counts of species in abundances classes the likelihood function is

$$L(\theta) = \sum^{C} n_i \ln P_i$$

for i = 1, 2, 3, ...C, where C is the number of abundance classes, n_i is the number of species in abundance class i.

P_i is the probability attributed by the distribution model to the observation of one species in class i, which depends on the vector θ of free parameters of the distribution model:

$$P_i = \int_{L_i}^{U_i} F(x \mid \theta) dx$$

where F(x|theta) is the value of the probability density function for a cover value x, under parameter values fixed at theta, and L_i and U_i are the lower and upper limits of the cover class i.

See `fitsad` for descriptions of each distribution model.

Value

An object of `fitsadC-class` which inherits from `mle2-class` and thus has methods for handling results of maximum likelihood fits from `mle2` and also specific methods to handle SADs models (see `fitsadC-class`).

Author(s)

Paulo I Prado <prado@ib.usp.br>, Murilo Dantas Miranda and Andre Chalom, after Ben Bolker, R Core Team.

Source

all fitting functions builds on `mle2` and methods from **bbmle** package (Bolker 2012), which in turn builds on `mle` function and associated classes and methods.

References

Bolker, B. and R Development Core Team 2012. bbmle: Tools for general maximum likelihood estimation. R package version 1.0.5.2. http://CRAN.R-project.org/package=bbmle

Chapman, D. G. 1956. Estimating the parameters of a truncated gamma distribution. *The Annals of Mathematical Statistics, 27(2)*: 498–506.

Magurran, A.E. 1989. *Ecological diversity and its measurement*. Princenton University Press.

Preston, F.W. 1948. The commonness and rarity of species. *Ecology 29*: 254–283.

Rinne, H. 2009. *The Weibull distribution: a handbook*. CRC Press

See Also

`dpareto`,for corresponding density functions created for fitting SADs; standard distributions dexp, dgamma, dlnorm, dweibull; `fitsadC-class`.

Examples

```
## An example using data from Vieira et al, see dataset "grasslands"
## Breakpoints of the abundance classes used (cover classes)
vieira.brk <- c(0,1,3,5,seq(15,100, by=10),100)
##  creates an histogram object
grass.h <- hist(grasslands$mids, breaks = vieira.brk, plot = FALSE)
#Fits Pareto, lognormal and gamma distributions
```

```
grass.p <- fitparetoC(grass.h)
grass.l <- fitlnormC(grass.h)
grass.g <- fitgammaC(grass.h)
## Predicted values by each model
grass.p.pred <- coverpred(grass.p)
grass.l.pred <- coverpred(grass.l)
grass.g.pred <- coverpred(grass.g)
## model selection
AICctab(grass.p, grass.l, grass.g, weights =TRUE, base = TRUE)
## A histogram with the densities predicted by each model
plot(grass.h, main = "", xlab = "Abundance (cover)")
## Adds predicted densities by each model to the plot
points(grass.p.pred, col = 1)
points(grass.l.pred, col = 2)
points(grass.g.pred, col = 3)
legend("topright", legend=c("Pareto","Log-normal", "Gamma"), col = 1:3, lty=1, pch =1)
```

---

fitsadC-class   *Class* "fitsadC" *for maximum likelihood fitting of species abundance distributions from data in abundance classes*

Description

This class extends mle2-class to encapsulate models of species abundance distributions (SADs) fitted by maximum likelihood, from data where species are classified in abundance classes (e.g, histograms or frequency tables of number of species in classes of abundances).

Objects from the Class

Objects can be created by calls of the form new("fitsadC", ...), or, more commonly a call to functions fitexpC, fitgammaC, fitlnormC, fitparetoC, fitweibullC, which fit a probability distribution to a table of frequency of species abundances.

Slots

sad: Object of class "character"; root name of the species abundance distribution fitted. See man page of fitsad for available models.

trunc: Object of class "numeric"; truncation value used in the fitted model. 'NA' for a non-truncated distribution.

hist: Object of class "histogram"; a table of frequencies of species in abundance classes, returned by the function hist.

call: Object of class "language"; The call to mle2.

call.orig: Object of class "language" The call to mle2, saved in its original form (i.e. without data arguments evaluated).

coef: Object of class "numeric"; Vector of estimated parameters.

fullcoef: Object of class "numeric"; Fixed and estimated parameters.

vcov: Object of class ″matrix″; Approximate variance-covariance matrix, based on the second derivative matrix at the MLE.

min: Object of class ″numeric″; Minimum value of objective function = minimum negative log-likelihood.

details: Object of class ″list″; Return value from [optim](#).

minuslogl: Object of class ″function″; The negative log-likelihood function.

method: Object of class ″character″; The optimization method used.

data: Object of class ″data.frame″; Data with which to evaluate the negative log-likelihood function.

formula: Object of class ″character″; If a formula was specified, a character vector giving the formula and parameter specifications.

optimizer: Object of class ″character″; The optimizing function used.

Extends

Class ["mle2"](#), directly.

Methods

coverpred signature(object = ″fitsadC″, sad = ″missing″, coef = ″missing″, trunc = ″missing″,breaks = ″missing″, mids = ″missing″, S = ″missing″): predicted number of species in each abundance class see [coverpred](#)

nobs signature(object = ″fitsadC″): Displays number of observations (number of species) in the data to which the model was fitted.

plot signature(x = ″fitsadC″, y = ″ANY″): diagnostic plots of the fitted model.

ppsad signature(x = ″fitsadC″, sad = ″missing″, coef = ″missing″, trunc = ″missing″): plot of observed vs predicted percentiles of the abundance distribution, details in [ppsad](#).

qqsad signature(x = ″fitsadC″, sad = ″missing″, coef = ″missing″, trunc = ″missing″, distr = ″missing″): plot of observed vs predicted quantiles of the abundance distribution, details in [qqsad](#).

radpred signature(object = ″fitsadC″, sad = ″missing″,rad = ″missing″, coef = ″missing″, trunc = ″missing″,distr = ″missing″, S = ″missing″, N = ″missing″): expected abundances of the 1st to n-th most abundant species, see [rad](#) and [radpred](#).

show signature(object = ″fitsadC″): Displays object.

Note

Class fitsadC only adds three slots to class mle2. The descriptions of slots inherited from mle2-class replicate those in [mle2-class](#).

Author(s)

Paulo I Prado <prado@ib.usp.br>, after Ben Bolker and R Core Team.

Source

this class builds on mle2-class of **bbmle** package (Bolker 2012), which in turn builds on mle-class.

References

Bolker, B. and R Development Core Team 2012. bbmle: Tools for general maximum likelihood estimation. R package version 1.0.5.2. http://CRAN.R-project.org/package=bbmle

See Also

mle2-class for all methods available from which fitsadC-class inherits; fitsadC for details on fitting SADs models from frequency tables; coverpred to get frequencies of species in abundances classes predicted from fitted models.

Examples

```
## Example of fitting a sad model to cover data
## Abundance classes: cover scale for plants
Lbrk <- c(0,1,3,5,15,25,35,45,55,65,75,85,95,100)
## To fit a sad model to cover data, data sould be in histogram format
grass.h <- hist(grasslands$mids, breaks = Lbrk, plot = FALSE)
class(grass.h) ## class "histogram"
## Fits a Pareto distribution to the histogram object
grass.p <- fitparetoC(grass.h)
class(grass.p)
## The class has a plot method to show diagnostic plots
par(mfrow=c(2,2))
plot(grass.p)
par(mfrow=c(1,1))
## Some methods inherited form mle2-class
summary(grass.p)
coef(grass.p)
AIC(grass.p)
```

---

grasslands   *Coverages of plants species in a plot in Southern Brazilian Grasslands*

Description

Coverage class of each plant species recorded in one of the plots set in grassland in Southern Brazil ('Pampa') by Vieira & Overbeck (2020).

Usage

```
data(grasslands)
```

Format

A data-frame with one plant species per row, and three vectors:

class cover class as coded in the original data set.

cover interval of the cover class for each plant species. Cover is the proportion of the area of the plot covered by all individuals of each species

upper Upper limit of the cover class of each plant species

mids Mid-point of the cover class for each plant sepcies

Source

Vieira & Overbeck (2020) provide cover data for each individual species in the whole set of eighty plots of 1 m2. The data frame grasslands corresponds to data from plot 'CA8', which has the largest number of species recorded among these plots.

References

Vieira, Mariana; Overbeck, Gerhard (2020). Small seed bank in grasslands and tree plantations in former grassland sites in the South Brazilian highlands [Dataset]. Dryad. https://doi.org/10.5061/dryad.n2z34tmsw

---

ICtab	*Compute table of information criteria and auxiliary info*

Description

This functions reimplement some functionality from package bbmle to work better with objects from the classes fitsad-class and fitrad-class.

Methods

AIC signature("mle2"): Akaike information criterion.

AICc signature("mle2"): Akaike information criterion corrected for small samples.

nobs signature(object = "fitsad"): Number of observations.

nobs signature(object = "fitrad"): Number of observations.

Author(s)

Paulo I Prado <prado@ib.usp.br>, Andre Chalom and Murilo Dantas Miranda, after Ben Bolker and R Core Team.

See Also

mle2-class for all methods available from which fitsad-class inherits; fitsad for details on fitting SADs models.

---

likelregions-class   *Class* "likelregions" *for likelihood profiles of maximum likelihood fits*

Description

This class provides a list of likelihood intervals for the parameters of a maximum likelihood fit. See the help on the function `likelregions` for details.

Examples

```
x1.gamma <- fitgamma(moths)
x1.p <- profile(x1.gamma)
likelregions(x1.p)
# Compare with...
confint(x1.p)
```

---

moths   *Moths caught with light traps at Rothamsted 1933-1936*

Description

Number of captured individuals of species of nocturnal macrolepidoptera (moths) in light traps at the Rothamsted Experimental Station, UK between 1933 and 1936.

Usage

```
data(moths)
```

Format

A unnamed vector of 240 integers.

Details

This is one of the datasets that C. B. Williams used to show the application of Fisher's log-series to describe the abundance of species in a random sample of a biological community.

Source

Column I of table 3 of Fisher et al. (1943), which did not provide species names.

References

Fisher, R.A., Corbert, A.S. and Williams, C.B. 1943. The Relation between the number of species and the number of individuals in a random sample of an animal population. *The Journal of Animal Ecology, 12*: 42–58.

---

octav-class   *Class* "octav" *for frequencies in abundance octaves*

Description

Data frame of frequencies of entities (usually species) in classes of logarithm of abundances at base 2 (Preston's octaves).

Usage

```
## S4 method for signature 'octav'
lines(x, prop=FALSE, mid=TRUE, ...)
## S4 method for signature 'octav'
plot(x, prop=FALSE, x.oct=FALSE, par.axis=list(), ...)
## S4 method for signature 'octav'
points(x, prop=FALSE, mid=TRUE, ...)
```

Arguments

x	an object of class octav
prop	logical; if TRUE relative frequencies are returned.
x.oct	logical; if TRUE axis labels are octave numbers, if FALSE upper limit of abundance class are used as labels.
mid	logical; if TRUE x coordinates of abundances are set to the mid of each octave, if FALSE x coordinates of abundances are the values of its octave (see examples)
par.axis	list; further graphical parameters for the plot axes.
...	further parameters to be passed to lines, points or plot functions (except axes in plot, which are set by par.axis.

Objects from the Class

Objects can be created by calls of the form new("octav", ...), but most often by a call to [octav](#) or [octavpred](#).

Slots

.Data: Object of class "list" a data frame of three vectors: octave number (integer), which is the upper limit of the class in log2; upper limit of the class in arithmetic scale (numeric); number of cases in each class (numeric).

names: Object of class "character"; names of the three vectors of .Data, "octave", "upper", and "Freq", respectively.

row.names: Object of class "data.frameRowLabels"; default line names for .Data.

.S3Class: Object of class "character"; indicates inheritance from S3 class data.frame.

Extends

Class `"data.frame"`, directly. Class `"list"`, by class "data.frame", distance 2. Class `"oldClass"`, by class "data.frame", distance 2. Class `"vector"`, by class "data.frame", distance 3.

Methods

lines `signature(x = "octav")`: adds frequency data contained in the object as lines in an octave plot created by `plot` method.

plot `signature(x = "octav", y = "ANY")`: creates a histogram of frequencies of species in each octave in the object, a.k.a 'Preston plot'

points `signature(x = "octav")`: adds frequency data contained in the object as points in a octave plot created by `plot` method.

Author(s)

Andre Chalom and Paulo I Prado <prado@ib.usp.br>

References

Magurran, A.E. 1989. *Ecological diversity and its measurement*. Princenton University Press.

Preston, F.W. 1948. The commonness and rarity of species. *Ecology 29*: 254–283.

See Also

[octav](#) to get an object of the class from a vector of abundances; [octavpred](#) to get an octav object of predicted abundances from a theoretical distribution; man page of `prestonfit` in package vegan for a detailed account of Preston's octaves and an alternative way to get octaves and model fitting.

Examples

```
## Creates an octav object from an abundance vector
birds.oc <- octav(birds)
moths.oc <- octav(moths)

## default plot
plot(birds.oc)
## Octave values instead of abundances at x-axis
plot(moths.oc, x.oct=TRUE)

## Using line and argument prop to superpose two data sets
## (Fisher's et al moth data and Preston's bird data)
## mid=FALSE plots points at each octave value and not
## in the midclass (default, useful for histograms)
plot(moths.oc, col=NULL, border=NA, prop=TRUE, x.oct=TRUE, xlab="Octave")
lines(moths.oc, prop=TRUE,  mid=FALSE)
lines(birds.oc, prop=TRUE, mid=FALSE, col="red")
```

octav-methods	*Frequencies of species in octaves*

Description

Creates an object of octav-class with number of species in octaves of abundances from a vector of abundances or from a fitted model.

Usage

```
octav(x, oct, preston=FALSE)
```

Arguments

x	a numerical vector of abundances or an object of class fitsad or fitrad.
oct	integer vector; the octaves to tabulate abundances. Should include all abundance values in x.
preston	logical; if 'TRUE' use Preston method to count frequencies (see details), if 'FALSE' class intervals are open on the left (default in cut).

Details

Preston (1948) popularized the use of histograms with logarithmic classes to depict species abundance distributions (Magurran 1989). Preston used classes at log base two, which he called 'octaves' as their end-points double from one class to the other. In Preston original method half of the species with abundances equal to the limits of octaves are credited to the neighboring octave. If preston=TRUE this non-standard method of class closure is applied. In general this makes the histogram more bell-shaped, as Preston expected (see example).

Value

an object of class octav, which is a data frame with three vectors:

octav	integer; octave number, which is the upper limit of the class in log2.
upper	numeric; upper limit of the class in arithmetic scale.
Freq	integer or numeric; (relative) frequencies of species in each class.

Methods

signature(x = "numeric") number of species in each octave in a vector of species abundances.

signature(x = "fitsad") number of species in each octave in the original abundance vector used to fit a model with fitsad.

signature(x = "fitrad") number of species in each octave in the original abundance vector used to fit a model with fitrad.

Author(s)

Paulo I Prado <prado@ib.usp.br>, Andre Chalom and Murilo Dantas Miranda

References

Magurran, A.E. 1989. *Ecological diversity and its measurement*. Princeton University Press.

Preston, F.W. 1948. The commonness and rarity of species. *Ecology 29*: 254–283.

See Also

[octav-class](#) for methods to create an octave plot; [octavpred](#) to get an octav object of predicted abundances from a theoretical distribution; [fitsad-class](#) and [fitrad-class](#) objects, from which you can also get an object of class octav; man page of prestonfit in package **vegan** for a detailed account of Preston's octaves and an alternative way to get octaves and fitting of species abundances distributions.

Examples

```
## BCI tree data
(bci.oc1 <- octav(bci, preston=TRUE))
## Comparing with standard class closure
par(mfrow=c(1,2))
plot(octav(bci), main="octav(bci, preston=FALSE)")
plot(bci.oc1, main="octav(bci, preston=TRUE)")
par(mfrow=c(1,1))
```

---

octavpred-methods   *Predicted frequencies of species in octaves*

Description

Creates an object of octav-class with the frequencies of species in octaves of abundances predicted by a species abundance distribution or by a rank-abundance distribution.

Arguments

object	an object of class fitrad or fitrad; fitted model of rank-abundance or species abundances distributions. Alternatively a numeric vector abundances of species.
sad, rad	character; root name of sad or rad distribution to calculate expected percentiles. See [fitsad](#) and [fitrad](#) for available distributions.
coef	named list of numeric values; parameter values of the distribution given in sad or rad. Parameters should be named as in the corresponding density function, and in the same order.
trunc	non-negative integer, trunc > min(x); truncation point if fitted distribution is truncated.

oct	integer vector; the octaves to tabulate abundances, see octav. If not provided, the octavpred method will generate an educated guess.
S	positive integer; number of species in the sample.
N	positive integer; number of individuals in the sample.
preston	logical. Should Preston original method be used? See octav.

Methods

signature(object = "fitrad", sad = "missing", rad = "missing", coef = "missing", trunc = "missing", oct = "A
number of species in each octave predicted from a rank-abundance model fitted with function
fitrad.

signature(object = "fitsad", sad = "missing", rad = "missing", coef = "missing", trunc = "missing", oct = "A
number of species in each octave predicted from an abundance distribution model fitted with
function fitsad.

signature(object = "missing", sad = "character", rad = "missing", coef = "list", trunc = "ANY", oct = "ANY",
number of species in each octave predicted from an abundance distribution named by sad with
parameters defined in coef.

signature(object = "numeric", sad = "character", rad = "missing", coef = "list", trunc = "ANY", oct = "ANY",
same as previous method, but with S and N taken from a vector of abundances given by object.

signature(object = "missing", sad = "missing", rad = "character", coef = "list", trunc = "ANY", oct = "ANY",
number of species in each octave predicted from an rank-abundance distribution named by rad
with parameters defined in coef.

signature(object = "numeric", sad = "missing", rad = "character", coef = "list", trunc = "ANY", oct = "ANY",
same as previous method, but with S and N taken from a vector of abundances given by object.

Author(s)

Paulo I Prado <prado@ib.usp.br>, Murilo Dantas Miranda and Andre Chalom.

See Also

octav and octav-class for generic function and methods to create an octave plot and details on
abundance octaves; fitsad-class and fitrad-class objects, from which you can also get an
object of class octav with observed and predicted values; man page of prestonfit in package
vegan for a detailed account of Preston's octaves and an alternative way to get octaves and fitting
of species abundances distributions.

Examples

```
## Predicted frequencies from a fitted model
## meta-community zero-sum multinomial for BCI data
bci.mzsm <- fitsad(bci, "mzsm")
bci.mzsm.oc <- octavpred(bci.mzsm)
## Preston plot with observed and predicted frequencies
plot(octav(bci))
lines(bci.mzsm.oc)
## Alternative model: local zero-sum multinomial
## Alonso & Mckane (Ecol. Lett. 2004, table 1) give theta = 44 and m = 0.15
```

```
bci.lzsm.oc <- octavpred( bci, sad = "volkov", coef =list(theta = 44, m = 0.15, J=sum(bci)) )
## Adding predicted frequencies to the plot
lines(bci.lzsm.oc, col = "red")
```

---

okland   *Abundances of land snail species in Norway*

Description

Number of individuals of 21 species of land snails caught in the vegetation in Norway, 1927.

Usage

```
data(okland)
```

Format

A unnamed vector of 21 integers.

Details

This is one of the data sets that Isao Motomura used to demonstrate his 'law of geometric series' of the abundance of species in a sample of a biological community. The abundance values and fitted series are in figure 2 of Motomura (1932).

Source

Second column ('A') of table 5 of Okland (1930).

References

Motomura, I. 1932. On the statistical treatment of communities (in Japanese). *Zool. Mag. (Tokyo) 44*: 379–383. English translation in Appendix I in Doi, H. and Mori, T. 2012. The discovery of species-abundance distribution in an ecological community. *Oikos 122*: 179–182.

Okland, F. 1930. Quantitative Untersuchungen der Landschneckenfauna Norwegens. I. *Zoomorphology 16:* 748–804.

---

plotprofmle-methods   *Log-likelihood profiles at original scale*

Description

Given a likelihood profile of a model (object of the class `profile.mle` or `profile.mle2`), the function `plotprofmle` plots the relative log-likelihood profiles and the plausibility intervals for each one of the (or selected ones) parameters of a model. These same plausibility regions might be returned by `likelregions`.

Usage

```
plotprofmle(object, nseg=20, ratio=log(8), which=NULL, ask=NULL,
    col.line="blue", varname=NULL, ...)
likelregions(object, nseg=100, ratio=log(8), ...)
```

Arguments

`object`	list of profile data; object of class `mle2` or `profile.mle2`.
`nseg`	positive integer; number of segments used by `spline` to interpolate the line of log-likelihood profile
`ratio`	real positive; log-likelihood ratio that defines the likelihood interval to be shown in the plot by `plotprofmle` or returned by `likelregions`. Set to `NULL` to suppress intervals from being displayed.
`which`	vector of positive integers; if a subset of profiles is required, the indexes of the mle's in `profobj` to be plotted.
`ask`	logical; if `TRUE`, the user is _ask_ed before each plot, see par(ask=.)
`col.line`	name; line color for the plausibility interval.
`varname`	vector of names; labels for the x-axis. If NULL defaults the names of mle's in `profobj`.
`...`	further arguments to be passed to `plot`.

Details

Log-likelihood profile plots are the basic diagnostic for model fitting by maximum likelihood methods. The profiles show the minimum of the log-likelihood function for a given value of a focal parameter, near the maximum likelihood estimate (mle) of this parameter. Profile objects in R (classes `profile.mle` and `profile.mle2`) return transformed values of the likelihood function, which are based on the deviance (=minus twice log-likelihood). These values are called 'z' and are the signed square-root of the deviance difference from the minimum deviance. As samples get larger, z-profiles tends to be symmetrical V-shaped, and are used to calculate confidence intervals using an approximation to the Chi-square distribution (see details in Bolker (2008) and in the **bbmle** vignette (`vignette('mle2',package='bbmle')`)).

In its original form (e.g. Edwards 1972), likelihood profiles do not use z-transformed values, and can be interpreted directly, even if they are asymmetric. At the scale of the log-likelihood function,

all values of the parameters resulting in a negative log-likelihood less or equal to a given value k are exp(k) times as plausible as the mle. Hence, exp(k) is a likelihood ratio, and delimits a plausibility interval (or likelihood interval) for the mle's.

Function `plotprofmle` plots profiles of the negative log-likelihood functions, along with the limits of likelihood interval for a given log-likelihood `ratio`.

Function `likelregions` returns the limits of the likelihood intervals for each parameter. This might be seen as an analog function for `confint`, and will return very similar values for corresponding ratios if the profile is symmetric and monotonic. However, if the profile is ill-behaved, `likelregions` might return more than one interval for each parameter, whereas `confint` will return NA with a warning.

Methods

plotprofmle `signature(object="profile.mle2")`:The preferred invocation for these methods.

plotprofmle `signature(object="mle2")`:A convenience wrapper that calls `profile` on the mle2 object and runs the former method.

likelregions `signature(object="profile.mle2")`:The preferred invocation for these methods.

likelregions `signature(object="mle2")`:A convenience wrapper that calls `profile` on the mle2 object and runs the former method.

Author(s)

João L.F. Batista, Andre Chalom, Paulo I. Prado <prado@ib.usp.br>

References

Bolker, B. 2008. *Ecological Models and Data in R*. Princeton: Princeton University Press.

Edwards, A.W.F. 1972. *Likelihood – An Account of the Statistical Concept of Likelihood and its Application to Scientific Inference*. New York: Cambridge University Press.

Royall, R.M. 2000. *Statistical Evidence: A Likelihood Paradigm*. London: Chapman and Hall.

See Also

`profile.mle.class`, `mle`, `mle-class` from **stats**; `profile.mle2.class`, `mle2`, `mle2-class` from **bbmle** package.

Examples

```
birds.pln <- fitsad(birds, "lnorm")
birds.pln.p <- profile(birds.pln)
par(mfrow=c(1,2))
plotprofmle(birds.pln.p)
par(mfrow=c(1,1))
likelregions(birds.pln.p)
# Compare with the confidence intervals
confint(birds.pln.p)
```

ppsad and pprad methods

Percentile-percentile plots for species-abundance and rank-abundance models

Description

Plots empirical percentiles vs corresponding theoretical values expected by a model for species abundances (SAD) or a model for species abundance ranks (RAD).

Usage

```
## S4 method for signature 'fitsad'
ppsad(x, plot=TRUE, line=TRUE, ...)
## S4 method for signature 'numeric'
ppsad(x, sad, coef, trunc=NA, plot=TRUE,
line=TRUE, ...)
## S4 method for signature 'fitrad'
pprad(x, plot=TRUE, line=TRUE, ...)
## S4 method for signature 'rad'
pprad(x, rad, coef, trunc=NA, plot=TRUE, line=TRUE, ...)
## S4 method for signature 'numeric'
pprad(x, rad, coef, trunc=NA, plot=TRUE, line=TRUE, ...)
```

Arguments

x	a numeric vector of abundances of species or a fitted sad/rad model (object of `fitsad-class` or `fitrad-class`, respectively). For pprad this argument can be also a rank-abundance table of rad-class.
sad, rad	character; root name of sad or rad distribution to calculate expected percentiles. See `fitsad` and `fitrad` for available distributions.
coef	named list of numeric values; parameter values of the distribution given in `sad` or `rad`. Parameters should be named as in the corresponding density function, and in the same order
trunc	non-negative integer, trunc > min(x); truncation point to fit a truncated distribution.
plot	logical; if 'TRUE' a percentile-percentile plot is produced. If not, only a data frame with theoretical and empirical values for percentiles of the data is invisibly returned.
line	logical; if 'TRUE' and plot is 'TRUE', the equivalence line y=x with `abline(0,1)` is added to the plot. If not, no line is drawn.
...	further arguments to be passed to the `plot` function.

Methods

ppsad signature(x = "fitsad", sad = "missing", coef = "missing", trunc = "missing", plot="ANY", line="ANY"): quantile-quantile plot for a fitted model of species abundances (a [fitsad-class](#) object). Only argument x should be provided.

ppsad signature(x = "numeric", sad = "character", coef = "list", trunc = "ANY", plot="ANY", line="ANY"): quantile-quantile plot of a numeric vector of abundances (x) vs a species abundance distributions defined by the following arguments.

pprad signature(x = "fitrad", rad = "missing", coef = "missing", trunc = "missing", plot="ANY", line="ANY"): quantile-quantile plot for a fitted model of species abundances ranks (a [fitrad-class](#) object). Only argument x should be provided.

pprad signature(x = "rad", rad = "character", coef = "list", trunc = "ANY", plot="ANY", line="ANY"): quantile-quantile plot of a table of abundance ranks (x) vs a species rank-abundance distribution defined by the following arguments.

pprad signature(x = "numeric", rad = "character", coef = "list", trunc = "ANY", plot="ANY", line="ANY"): quantile-quantile plot of a numeric vector of abundances (x) vs a species rank-abundance distribution defined by the following arguments.

Author(s)

Paulo I Prado <prado@ib.usp.br> and Murilo Dantas Miranda.

References

Thas, O. 2010. Comparing distributions. Springer.

Examples

```
## An example with SADs
data(moths)
## fits log-series distribution to abundance data
moths.ls <- fitsad(moths, "ls")
## fits lognormal distribution truncated at 0.5
moths.ln <- fitsad(moths,"lnorm", trunc=0.5)
## Plots with the model object and with abundance vector
par(mfrow=c(2,2))
ppsad(moths.ls)
ppsad(moths, sad="ls", coef=as.list(coef(moths.ls)) )
ppsad(moths.ln)
ppsad(moths, sad="lnorm", coef=as.list(coef(moths.ln)), trunc=0.5)
par(mfrow=c(1,1))

## An example with RADs
data(okland)
## Fits broken-stick RAD model
ok.bs <- fitrad(okland, "rbs")
## Fits geometric series RAD model
ok.gs <- fitrad(okland, "gs")
## Plots with the model object and with the abundance vector
par( mfrow=c(2, 2) )
```

```
pprad(ok.bs)
pprad(okland, rad="rbs", coef=as.list(coef(ok.bs)))
pprad(ok.gs)
pprad(okland, rad="gs", coef=as.list(coef(ok.gs)))
par(mfrow=c(1,1))
```

---

pred.logser   *Predicted number of species by Fisher's Logseries*

Description

Given a vector of species abundances, Fisher's alpha and total number of species and individuals in a sample, returns the number of species for each abundance value expected by the Fisher's logseries model

Usage

```
pred.logser(x, alpha, J, S)
```

Arguments

x	Vector of (non-negative integer) abundances of species in a sample.
alpha	Fisher's alpha, the single parameter of log-series.
J	Total number of individuals in the sample.
S	Total number of species in the sample.

Details

The Fisher logseries is a limiting case of the Negative Binomial where the dispersion parameter of the negative binomial tends to zero. It was originally proposed by Fisher (1943) to relate the expected number of species in a sample from a biological community to the sample size as:

$S = alpha * \log(1 + J/alpha)$

Where alpha is the single parameter of the logseries distribution, often used as a diversity index. From this relation follows that the expected number of species with x individuals in the sample is

$S(x) = alpha*X^x/x$

Where X is a function of alpha and J, that tends to one as the sample size J increases:

$X = J / (alpha + J)$

Since the logseries model is a function that relates S to J using a single parameter (alpha), once two of these quantities are known the remaining is determined. So the function allows the input of any two among S, J and alpha. If the user does not provide at least two of these values, an error message is returned.

This function returns the expected number of species with abundance x, which is

$E[S(x)] = x^{(-1)}*alpha*X^x$

Value

A (vector) of expected number of species to each abundance provided by argument x

Author(s)

Paulo I. Prado <prado@ib.usp.br>.

References

Pielou, E.C. 1977. *Mathematical Ecology*. New York: John Wiley and Sons.

Fisher, R.A, Corbert, A.S. and Williams, C.B. 1943. The Relation between the number of species and the number of individuals in a random sample of an animal population. *The Journal of Animal Ecology, 12(1)*: 42–58.

See Also

[dls](#) for the log-series distribution; and [fitls](#), fishers.alpha in package **untb** and fisherfit in package **vegan** for fitting the log-series to abundance data.

Examples

```
data(moths) # Willians' moth data
pred.logser(1:5, J=sum(moths), S=length(moths)) #predicted
table(moths)[1:5] # observed
```

---

qqsad and qqrad methods

Quantile-quantile plots for species-abundance and rank-abundance models

---

Description

Plots empirical quantiles vs corresponding theoretical values expected by a model for species abundances (SAD) or a model for species abundance ranks (RAD).

Usage

```
## S4 method for signature 'fitsad'
qqsad(x, plot=TRUE, line=TRUE, ...)
## S4 method for signature 'numeric'
qqsad(x, sad, coef, trunc=NA, distr, plot=TRUE,
line=TRUE, ...)
## S4 method for signature 'fitrad'
qqrad(x, plot=TRUE, line=TRUE, ...)
## S4 method for signature 'rad'
qqrad(x, rad, coef, trunc=NA, plot=TRUE, line=TRUE, ...)
## S4 method for signature 'numeric'
qqrad(x, rad, coef, trunc=NA, plot=TRUE, line=TRUE, ...)
```

Arguments

x	a numeric vector of abundances of species or a fitted sad/rad model (object of [fitsad-class](#) or [fitrad-class](#), respectively). For qqrad this argument can be also a rank-abundance table of [rad-class](#).
sad, rad	character; root name of sad or rad distribution to calculate expected percentiles. See [fitsad](#) and [fitrad](#) for available distributions.
coef	named list of numeric values; parameter values of the distribution given in sad or rad. Parameters should be named as in the corresponding density function, and in the same order
trunc	non-negative integer, trunc > min(x); truncation point to fit a truncated distribution.
distr	Deprecated since sads 0.2.4. See [distr](#) function
plot	logical; if TRUE a percentile-percentile plot is produced. If not, only a data frame with theoretical and empirical values for percentiles of the data is invisibly returned.
line	logical; if TRUE and plot is TRUE, the equivalence line y=x with abline(0,1) is added to the plot. If not, no line is drawn.
...	further arguments to be passed to the plot function.

Methods

qqsad signature(x = "fitsad", sad = "missing", coef = "missing", trunc = "missing", distr = "missing", plot="ANY", line="ANY"): quantile-quantile plot for a fitted model of species abundances (a [fitsad-class](#) object). Only argument x should be provided

qqsad signature(x = "numeric", sad = "character", coef = "list", trunc = "ANY", distr = "ANY", plot="ANY", line="ANY"): quantile-quantile plot of a numeric vector of abundances (x) vs a species abundance distributions defined by the following arguments.

qqrad signature(x = "fitrad", rad = "missing", coef = "missing", trunc = "missing", plot="ANY", line="ANY"): quantile-quantile plot for a fitted model of species abundances ranks (a [fitrad-class](#) object). Only argument x should be provided

qqrad signature(x = "rad", rad = "character", coef = "list", trunc = "ANY", plot="ANY", line="ANY"): quantile-quantile plot of a table of abundance ranks (x) vs a species rank-abundance distribution defined by the following arguments.

qqrad signature(x = "numeric", rad = "character", coef = "list", trunc = "ANY", plot="ANY", line="ANY"): quantile-quantile plot of a numeric vector of abundances (x) vs a species rank-abundance distribution defined by the following arguments.

Author(s)

Paulo I Prado <prado@ib.usp.br> and Murilo Dantas Miranda.

References

Thas, O. 2010. Comparing distributions. Springer.

Examples

```
## An example with SADs
data(moths)
## fits log-series distribution to abundance data
moths.ls <- fitsad(moths, "ls")
## fits lognormal distribution truncated at 0.5
moths.ln <- fitsad(moths,"lnorm", trunc=0.5)
## Plots with the model object and with abundance vector
par(mfrow=c(2,2))
qqsad(moths.ls)
qqsad(moths, sad="ls", coef=as.list(coef(moths.ls)))
qqsad(moths.ln)
qqsad(moths, sad="lnorm", coef=as.list(coef(moths.ln)), trunc=0.5)
par(mfrow=c(1,1))

## An example with RADs
data(okland)
## Fits broken-stick RAD model
ok.bs <- fitrad(okland, "rbs")
## Fits geometric series RAD model
ok.gs <- fitrad(okland, "gs")
## Plots with the model object and with the abundance vector
par( mfrow=c(2, 2) )
qqrad(ok.bs)
qqrad(okland, rad="rbs", coef=as.list(coef(ok.bs)))
qqrad(ok.gs)
qqrad(okland, rad="gs", coef=as.list(coef(ok.gs)))
par(mfrow=c(1,1))
```

---

rad   *Rank-abundance data frame*

Description

Creates a data frame with ranked abundances of `rad-class` from a vector of abundances, histogram, or fitted model object.

Usage

```
rad(x)
```

Arguments

x   a numerical vector of abundances or an histogram, or an object of class `fitsad`,
                `fitrad` or `fitsadC`.

Value

an object of [rad-class](), which is a data frame with two vectors:

rank	integer; abundance rank of each species (integer), from most abundant (rank=1) to the least abundant (rank=length(rank))
abund	numeric; abundance of each species

Author(s)

Paulo I. Prado <prado@ib.usp.br> and Murilo Dantas Miranda.

References

Whittaker, R. H. 1965, Dominance and Diversity in Land Plant Communities. *Science, 147*: 250–260.

See Also

[rad-class]() for methods to create a rank-abundance data frame; [radpred]() to get a rad-class object of predicted abundances from a theoretical distribution; [fitsad-class]() , [fitsadC-class]() and [fitrad-class]() objects, from which you can also get an object of class rad; [qqrad]() for quantile-quantile plots from a rad-class object, and [pprad]() for percentile-percentile plots.

Examples

```
## rank-abundance plot
plot(rad(okland))
```

---

rad-class   *Class* "rad" *for rank-abundance data*

Description

Data frame of ranked abundances of species

Usage

```
## S4 method for signature 'rad'
lines(x, prop=FALSE, ...)
## S4 method for signature 'rad'
plot(x, prop=FALSE, ...)
## S4 method for signature 'rad'
points(x, prop=FALSE, ...)
```

Arguments

x	an object of class rad
prop	logical; if TRUE relative abundances are returned.
...	further parameters to be passed to lines, points or plot functions.

Objects from the Class

Objects can be created by calls of the form new("rad", ...), but most often by a call to rad or radpred.

Slots

.Data: Object of class "list"; a data frame of two vectors: abundance rank of each species (integer), from most abundant (rank=1) to the least abundant (rank=length(rank)); abundance of each species (numeric)

names: Object of class "character"; names of the two vectors of .Data, "rank" and "abund", respectively.

row.names: Object of class "data.frameRowLabels"; default line names for .Data; integer indexes from 1 to nrow(.Data).

.S3Class: Object of class "character"; indicates inheritance from S3 class data.frame.

Extends

Class "data.frame", directly. Class "list", by class "data.frame", distance 2. Class "oldClass", by class "data.frame", distance 2. Class "vector", by class "data.frame", distance 3.

Methods

lines signature(x = "rad"): adds rank-abundance data contained in the object as lines in rank-abundance plots created by plot method.

plot signature(x = "rad", y = "ANY"): creates a rank-abundance plot from data in the object.

points signature(x = "rad"): adds rank-abundance data contained in the object as points in rank-abundance plots created by plot method.

pprad signature(x = "rad", rad = "character", coef = "list"): percentile-percentile plot, see pprad.

qqrad signature(x = "rad", rad = "character", coef = "list", trunc = "ANY"): quantile-quantile plot, see qqrad.

Author(s)

Paulo I. Prado <prado@ib.usp.br> and Murilo Dantas Miranda

References

Whittaker, R. H. 1965, Dominance and Diversity in Land Plant Communities. *Science, 147*: 250–260.

See Also

[rad](#) to get an object of the class from a vector of abundances or from an histogram; [radpred](#) to get a rad-class object of predicted abundances from a theoretical distribution, [qqrad](#) for quantile-quantile plots from a rad-class object, and [pprad](#) for percentile-percentile plots.

Examples

```
## Creates a rad object from a vector of abundances
birds.rad <- rad(birds)

## Rank-abundance plot
plot(birds.rad)

## Same, with non-default graphical parameters
plot(birds.rad, pch=19, xlab="Abundance rank of species")

## Adding points from another data set
points(rad(okland), pch=19)
```

---

radpred-methods	*Predicted ranked abundance of species*

Description

Creates an object of rad-class with the ranked abundances predicted by a species abundance distribution or a rank-abundance distribution.

Arguments

object	an object of class fitrad or fitsad; fitted model of rank-abundance or species abundances distributions. Alternatively a numeric vector abundances of species.
sad, rad	character; root name of sad or rad distribution to calculate expected percentiles. See [fitsad](#) and [fitrad](#) for available distributions.
coef	named list of numeric values; parameter values of the distribution given in sad or rad. Parameters should be named as in the corresponding density function, and in the same order.
trunc	non-negative integer, trunc > min(x); truncation point if fitted distribution is truncated.
distr	Deprecated since sads 0.2.4. See [distr](#) function
S	positive integer; number of species in the sample.
N	positive integer; number of individuals in the sample.

Methods

signature(object = "fitrad", sad = "missing", rad = "missing", coef = "missing", trunc = "missing", distr =
    ranked abundances of species predicted from a rank-abundance model fitted with function
    fitrad.

signature(object = "fitsad", sad = "missing", rad = "missing", coef = "missing", trunc = "missing", distr =
    ranked abundances of species predicted from a abundance distribution model fitted with func-
    tion fitsad.

signature(object = "fitsadC", sad = "missing", rad = "missing", coef = "missing", trunc = "missing", distr =
    ranked abundances of species predicted from a abundance distribution model fitted with func-
    tion fitsadC.

signature(object = "missing", sad = "character", rad = "missing", coef = "list", trunc = "ANY", distr = "ANY
    ranked abundances of species predicted from abundance distribution named by sad with pa-
    rameters defined in coef.

signature(object = "numeric", sad = "character", rad = "missing", coef = "list", trunc = "ANY", distr = "ANY
    same as previous method, but with S and N taken from a vector of abundances given by object.

signature(object = "missing", sad = "missing", rad = "character", coef = "list", trunc = "ANY", distr = "mis
    ranked abundances of species predicted from a rank-abundance distribution named by rad
    with parameters defined in coef.

signature(object = "numeric", sad = "missing", rad = "character", coef = "list", trunc = "ANY", distr = "mis
    same as previous method, but with S and N taken from a vector of abundances given by object.

Note

The rank-abundance function is the inverse of the quantile function of the species abundance distri-
bution (May 1975). radpred uses numeric interpolation with approxfun to find quantiles, instead
of the slower bisection method used in many quantile functions in the package **sads**. Hence results
from radpred and from the quantile functions may not match exactly.

Author(s)

Paulo I. Prado <prado@ib.usp.br> and Murilo Dantas Miranda

References

May, R.M. 1975. Patterns of Species Abundance and Diversity. In Cody, M.L. and Diamond, J.M.
(Eds) *Ecology and Evolution of Communities*. Harvard University Press. pp 81–120.

Examples

```
## Predicted frequencies from a fitted model
## meta-community zero-sum multinomial for BCI data
moths.mzsm <- fitsad(moths, "mzsm")
moths.mzsm.r <- radpred(moths.mzsm)
## Rank-abundance plot with observed and predicted frequencies
plot(rad(moths))
lines(moths.mzsm.r)
## Alternative model: local zero-sum multinomial
## Alonso & Mckane (Ecol. Lett. 2004, table 1) give theta = 41 and m = 0.77
```

```
moths.lzsm.r <-
  radpred( moths, sad = "volkov",
          coef =list(theta = 41, m = 0.77, J=sum(moths))
        )
## Adding predicted frequencies to the plot
lines(moths.lzsm.r, col = "red")
```

---

rsad   *Sampling from a species abundance distribution*

Description

A given number of realizations of a probability distribution (species abundances in a community) is sampled with replacement by a Poisson or Negative Binomial process, or without replacement by a hypergeometric process.

Usage

```
rsad(S = NULL, frac, sad = c("bs","gamma","geom","lnorm","ls","mzsm","nbinom","pareto",
                        "poilog","power", "powbend", "volkov", "weibull")),
            coef, trunc=NaN, sampling=c("poisson", "nbinom", "hypergeometric"),
                k, zeroes=FALSE, ssize=1)
```

Arguments

S	positive integer; number of species in the community, which is the number of random deviates generated by the probability distribution given by argument sad. Notice that for some distributions, the value of S is deduced from the coefficients, so this value is ignored (see details).
frac	single numeric 0 < frac <= 1; fraction of the community sampled. Usually the proportion of total number of individuals or of total area or volume sampled. Notice that assumptions of Poisson and binomial sampling are only valid for small values of frac (see Details).
sad	numeric; a vector of positive real numbers depicting abundances of species in a community or sample OR character; root name of community sad distribution - e.g., lnorm for the lognormal distribution rlnorm; geom for the geometric distribution rgeom or ls for the lognormal distribution ls.
coef	list with named arguments to be passed to the probability function defined by the argument sad.
trunc	The truncation point at which the random distribution defined in argument sad should be truncated; see rtrunc. The default value of NaN means that no truncation is performed.
sampling	character; if poisson the sampling process is Poisson (independent sampling of individuals with replacement); if nbinom negative binomial sampling is used to simulate aggregation of individuals in sampling units with replacement; finally, hypergeometric samples a fixed number of frac times the number of individuals in the simulated community, without replacement. Partial matching is allowed.

k	positive; size parameter for the sampling binomial negative. This parameter is ignored for other sampling techniques.
zeroes	logical; should zero values be included in the returned vector?
ssize	positive integer; sample size: number of draws taken from the community.

Details

This function simulates one or more random samples taken from a community with S species. The expected species abundances in the sampled community can (i) follow a probability distribution given by the argument sad or (ii) be a numeric vector provided by the user through this same argument. A fraction frac of the whole set of units that made up the community (usually individuals) is sampled. Hence the expected abundance in the sample of each species is frac*n, where n is the species' expected abundance in the community.

Three sampling processes can be simulated. Sampling with replacement can be done with Poisson (individuals are sampled independently) or negative binomial sampling (where individuals of each species are aggregated over sampling units). The "hypergeometric" sampling scheme draws frac * n individuals without replacement.

For Poisson and negative binomial schemes the species abundances in the sample are statistically independent. In general terms, these two sampling schemes takes a Poisson or negative binomial sampling with replacement of a vector of S realizations of a random variable, with the sampling intensity given by frac. The resulting values are realizations of a Poisson (or a Negative Binomial) random variable where the parameter that corresponds to the mean (=expected value of the variable) follows a probability distribution or the numeric vector given by the argument sad. Because these two sampling schemes assume replacement but the sampled community is finite, they are valid only when the fraction of the sampled community is small (frac«1).

The "hypergeometric" scheme simulates a sample of a fixed total number of individuals from the community. Therefore, abundances of the species in the sample are interdependent (Connoly et al. 2009). Sampling is carried out with base::sample(..., replace = FALSE). This scheme samples without replacement a finite community and therefore provides valid results for any value of frac.

For the broken-stick, logseries, MZSM and Volkov distributions, the expected value of S is deduced from the coefficients provided in the argument coef; thus, the value of the parameter S is ignored and may be left blank. The expressions for the number of species in each case are:

* Broken-stick: coefficient S * Log-series: alpha log(1 + N/alpha) * MZSM: sum_x=1^J theta/x (1 - x/J)^(theta - 1) * Volkov: sum of the unnormalized PDF from 1 to J, see dvolkov

Value

if ssize=1 a vector of (zero truncated) abundances in the sample; if ssize>1 a data frame with sample identification, species identification, and (zero truncated) abundances.

Author(s)

Paulo I. Prado <prado@ib.usp.br> and Andre Chalom.

References

Pielou, E.C. 1977. *Mathematical Ecology*. New York: John Wiley and Sons.

Green, J. and Plotkin, J.B. 2007 A statistical theory for sampling species abundances. *Ecology Letters 10*:1037–1045

Connolly, S.R., Dornelas, M., Bellwood, D.R. and Hughes, T.P. 2009. Testing species abundance models: a new bootstrap approach applied to Indo-Pacific coral reefs. *Ecology, 90*(11): 3138–3149.

See Also

`dpoix`, `dpoig` and `dpoilog` for examples of compound Poisson probability distributions like those simulated by `rsad`.

Examples

```
##A Poisson sample from a community with a lognormal sad
samp2 <- rsad(S = 100, frac=0.1, sad="lnorm", coef=list(meanlog=5, sdlog=2))
## Preston plot
plot(octav(samp2))
## Once this is a Poisson sample of a lognormal community, the abundances
## in the sample should follow a Poisson-lognormal distribution.
## Adds line of theoretical Poisson-lognormal with
## mu=meanlog+log(frac) and sigma=sdlog)
## Predicted by the theoretical Poisson-lognormal truncated at zero
samp2.pred <- octavpred(samp2, sad="poilog", coef= list(mu=5+log(0.1), sig=2), trunc=0)
## Adding the line in the Preston plot
lines(samp2.pred)
```

---

summary.sads-class   *Summary for fitsad/fitrad calls*

Description

This function works almost exactly as bbmle's summary.mle2, but it includes a "fixed parameters" line for models with fixed parameters, such as `fitls` or fitvolkov.

Usage

```
## S4 method for signature 'summary.sads'
show(object)

## S4 method for signature 'fitsad'
summary(object)

## S4 method for signature 'fitrad'
summary(object)
```

Arguments

object   An object of class fitsad/fitrad is required to generate a summary.sads object.

---

trueLL-methods   *True likelihood for continuous variables*

Description

Calculates the corrected likelihood for independent observations of a continuous variable that follows a given (truncated) density function, given a measurement precision.

Arguments

object   vector or histogram of observed values, or an object of fitsad-class or fitrad-class

dist   character; root name of the continuous density distribution of the variable - e.g., lnorm for the lognormal distribution; gamma for the gamma distribution.

coef   named list; values of the coefficients of the continuous density distribution, in the same order and with the same names as in the probability function defined by the argument dist.

trunc   real number trunc <= min(x); value from which the density distribution is truncated. Currently only lower-tail truncation (or right-truncation) is implemented. If this argument is omitted the whole distribution is used (default).

dec.places   positive integer; number of decimal places used in the measurement of the observed values. Observed values will be rounded to this number of decimals. This argument defines the measurement precision, see details. If neither dec.places nor breaks is given, this defaults to zero.

breaks   A numeric vector giving the breakpoints between count classes. Either dec.places or breaks can be provided, but not both.

...   named arguments to be passed to the probability function defined by the argument dens.

Details

The (log)likelihood function is often defined as any function proportional to the (sum) product of (log) probability density of the observations. In its original formulation, however, the likelihood is proportional to the product of probabilities, not probabilities densities (Fisher 1922, stressed by Lindsey 1999). For continuous variables, these probabilities are calculated through integration of the probability density function in an interval. For observed values of continuous variables, this interval is defined by the measurement precision. The likelihood function is thus any function proportional to

prod (integral_(x-D)^(x+D) f(x) dx)

where x is the observed value, f(x) the density function and D is half the measurement precision:

D = 10^(-'dec.places')/2

For continuous variables aggregated in discrete classes, such as frequency tables of histograms, the probability of a given observation is

prod (integral_L^U f(x) dx)

where L and U are the lower and upper limits of the classes, as defined by argument `breaks`.

Because products of probabilities quickly tends to zero, probabilities are usually converted to their logs and then summed to give the log-likelihood function.

Methods

signature(object = "fitsad", dist = "missing", coef = "missing", trunc = "missing", dec.places = "ANY", bre
log-likelihood value of an abundance distribution model object fitted with function `fitsad`.

signature(object = "fitrad", dist = "missing", coef = "missing", trunc = "missing", dec.places = "ANY", bre
log-likelihood value of an abundance distribution model object fitted with function `fitrad`.

signature(x = "numeric", dist = "character", coef = "list", trunc = "ANY", dec.places = "ANY", breaks="ANY"
log-likelihood value of a vector of abundances object, fitted to a continuous distribution named by `dist` with parameters defined in `coef`.

signature(object = "histogram", dist = "character", coef = "list", trunc = "missing", dec.places = "ANY", b
log-likelihood value of the abundances given by histogram object, fitted to a continuous distribution named by `dist` with parameters defined in `coef`.

Note

WARNING: this function is extremely sensitive to the interval breaks provided (or to the decimal.places), which may result in surprising results. Use with great caution.

Author(s)

Paulo I. Prado <prado@ib.usp.br> and Andre Chalom

References

Fisher, R.A. 1922. On the mathematical foundations of theoretical statistics. *Phil. Trans. R. Soc. Lond. A 222*: 309–368.

Lindsey, J.K. 1999. Some statistical heresies. *The Statistician 48*(1): 1–40.

See Also

`logLik` in the package **MASS** and `logLik-methods` in package **bbmle**.

Examples

```
##Random sample of a lognormal distribution with precision = 0.1
x <- round(rlnorm(500, meanlog = 3, sdlog = 0.5), 1)
## Log-likelihood given by fitdistr
library(MASS)
logLik(fitdistr(x, "lognormal"))
## Which is the sum of log of densities
sum( dlnorm(x, meanlog=mean(log(x)), sdlog=sd(log(x)), log=TRUE) )
```

```
## Correct log-likelihood
trueLL(x, "lnorm", coef=list(meanlog=mean(log(x)), sdlog=sd(log(x))), dec.places=1)
# Alternative invocation
trueLL(fitlnorm(x))

## Data in classes
xoc <- octav(x)
xc <- as.numeric(as.character(xoc$octave))
xb <- 2^(c(min(xc)-1, xc))
xh <- hist(x, breaks=xb, plot=FALSE)
xll <- trueLL(xh, dist="lnorm", coef = list(meanlog=mean(log(x)), sd=sd(log(x))))
xp <- diff(plnorm(xh$breaks, mean(log(x)), sd(log(x))))
xll2 <- sum( rep(log(xp), xh$counts))
all.equal(xll, xll2) # should be TRUE
```

---

updatesad   *Updating of MLE fits by profiling*

Description

These functions update a fitsad/fitrad object by running the optimizer again starting on better fit
returned by profile.

These functions were not extensively tested, and should be considered in beta testing phase.

Usage

```
updatesad(object, ...)
updaterad(object, ...)
```

Arguments

object   object of the class [fitsad-class](#) or [fitrad-class](#) for which the optimization
                did not converge, or converged to a local minimum.

...   list of additional parameters to be passed to the fitsad or fitrad fitting pro-
                cedure. May include a different optimizer, method, or upper/lower bounds for
                the parameters. Refer to the fitsad and fitrad man pages for details of which
                parameters may be specified.

Details

The updatesad function runs a new profile of the fitted object, and if the profile is able to find a
better fit, it runs a new optimization starting on this better fit. If the profiling does not find a new fit,
the function exits with error. The actual processing is done by updatesad. updaterad is simply a
convenience alias.

Value

An object of [fitsad-class](#) or [fitrad-class](#).

Author(s)

Paulo I Prado <prado@ib.usp.br> and Andre Chalom

See Also

[fitsad-class](), [fitrad-class](), [fitsadC-class]()

Index