Package 'gambin'

October 13, 2022

Type Package

Title Fit the Gambin Model to Species Abundance Distributions

Version 2.5.0

Description Fits unimodal and multimodal gambin distributions to species-abundance distributions
from ecological data, as in in Matthews et al. (2014) <DOI:10.1111/ecog.00861>.
'gambin' is short for 'gamma-binomial'. The main function is fit_abundances(), which estimates
the 'alpha' parameter(s) of the gambin distribution using maximum likelihood. Functions are
also provided to generate the gambin distribution and for calculating likelihood statistics.

Imports stats, graphics, doParallel, gtools, foreach, parallel

Suggests testthat, knitr, rmarkdown

Depends R(>= 3.0.0)

URL https://github.com/txm676/gambin/

BugReports https://github.com/txm676/gambin/issues

License GPL-3

RoxygenNote 7.1.1

VignetteBuilder knitr

Encoding UTF-8

NeedsCompilation no

Author Thomas Matthews [aut, cre],
Michael Krabbe Borregaard [aut],
Karl Ugland [aut],
Colin Gillespie [aut]

Maintainer Thomas Matthews <txm676@gmail.com>

Repository CRAN

Date/Publication 2021-04-16 18:10:05 UTC

R topics documented:

---

gambin-package   *Fit the gambin model to species abundance distributions*

Description

This package provides functions for fitting unimodal and multimodal gambin distributions to species-abundance distributions from ecological data. The main function is `fit_abundances()`, which estimates the 'alpha' parameter(s) of the gambin distribution using maximum likelihood.

Details

The gambin distribution is a sample distribution based on a stochastic model of species abundances, and has been demonstrated to fit empirical data better than the most commonly used species-abundance models (see references). Gambin is a stochastic model which combines the gamma distribution with a binomial sampling method. To fit the gambin distribution, the abundance data is first binned into octaves. The expected abundance octave of a species is given by the number of successful consecutive Bernoulli trials with a given parameter p. The parameter p of species is assumed to distributed according to a gamma distribution. This approach can be viewed as linking the gamma distribution with the probability of success in a binomial process with x trials. Use the `fit_abundances()` function to fit the gambin model to a vector of species abundances, optionally using a subsample of the individuals. Use the `mult_abundances()` function to fit the gambin model to multiple sites / samples and return the alpha values for each model fit (both the raw values and the alpha values standardised by the number of individuals).The package estimates the alpha (shape) parameter with associated confidence intervals. Methods are provided for plotting the results, and for calculating the likelihood of fits.

The package now provides functionality to fit multimodal gambin distributions (i.e. a gambin distribution with more than one mode), and to deconstruct and examine a multimodal gambin model fit (`deconstruct_modes`).

References

Matthews, T.J., Borregaard, M.K., Ugland, K.I., Borges, P.A.V, Rigal, F., Cardoso, P. and Whittaker, R.J. (2014) The gambin model provides a superior fit to species abundance distributions with a single free parameter: evidence, implementation and interpretation. Ecography 37: 1002-1011.

Matthews, T.J., Borregaard, M.K., Gillespie, C.S., Rigal, F., Ugland, K.I., Krüger, R.F., Marques, R., Sadler, J.P., Borges, P.A.V., Kubota, Y. & Whittaker, R.J. (2019) Extension of the gambin model to multimodal species abundance distributions. Methods in Ecology and Evolution, 10, 432-437.

Ugland, K.I., Lambshead, F.J.D., McGill, B.J., Gray, J.S., O'Dea, N., Ladle, R.J. & Whittaker, R.J. (2007). Modelling dimensionality in species abundance distributions: description and evaluation of the Gambin model. Evolutionary Ecology Research, 9, 313-324.

See Also

https://github.com/txm676/gambin

Examples

```
data(moths, package = "gambin")
fit = fit_abundances(moths)
barplot(fit)
lines(fit)
AIC(fit)
```

AIC.gambin   *Likelihood statistics for the GamBin model*

Description

Uses likelihood and information theoretical approaches to reveal the degree of fit of the GamBin model to empirical species abundance distributions.

Usage

```
## S3 method for class 'gambin'
AIC(object, ...)

AICc(object, ...)

## S3 method for class 'gambin'
AICc(object, ...)

## S3 method for class 'gambin'
BIC(object, ...)

## S3 method for class 'gambin'
logLik(object, ...)
```

Arguments

`object`	An object of type `gambin`
`...`	Further arguments to pass to the function

Value

`logLik` returns an R object of type `logLik`. The other function return the numerical value of the statistic

References

Akaike, Hirotugu. "A new look at the statistical model identification." Automatic Control, IEEE Transactions on 19.6 (1974): 716-723.

Examples

```
data(moths)
fit = fit_abundances(moths)
AIC(fit)
```

---

categ   *Simulated bird SAD dataset with species classification data*

Description

A randomly generated bird SAD dataset where each species has been randomly classified according to its origin (native, exotic or invasive).

Format

A dataframe with three columns: 1) 'abundances' = the abundance of each species, 2) 'species' = the species names, and 3) 'status' the species origin classification. In regards to (3) each species is classified as either native (N), exotic (E) or invasive (I).

Source

This package.

Examples

```
data(categ, package = "gambin")
```

---

create_octaves	*Reclassify a vector of species' abundances into abundance octaves*

Description

Creates abundance octaves by a log2 transform that doubles the number of abundance classes within each octave (method 3 of Gray, Bjoergesaeter & Ugland 2006). Octave 0 contains the number of species with 1 individual, octave 1 the number of species with 2 or 3 individuals, octave 2 the number of species with 4 to 7 individuals, and so forth.

Usage

```
create_octaves(abundances, subsample = 0)
```

Arguments

abundances   A numerical vector of species abundances in a community.

subsample   If > 0, the community is subsampled by this number of individuals before creating octaves. This is useful for analyses where alpha is estimated from a standardized number of individuals.

Value

A data.frame with two variables: octave with the name of each octave and species with the number of species in that octave.

References

Gray, J.S., Bjoergesaeter, A. & Ugland, K.I. (2006) On plotting species abundance distributions. Journal of Animal Ecology, 75, 752-756.

Examples

```
data(moths)
create_octaves(moths)
```

---

deconstruct_modes	*Deconstruct a multimodal gambin model fit*

Description

Deconstruct a multimodal gambin model fit by locating the modal octaves and (if species classification data are provided) determining the proportion of different types of species in each octave.

Usage

```
deconstruct_modes(
  fit,
  dat,
  peak_val = NULL,
  abundances = "abundances",
  species = "species",
  categ = NULL,
  plot_modes = TRUE,
  col.statu = NULL,
  plot_legend = TRUE,
  legend_location = "topright"
)
```

Arguments

fit	A gambin model fit where the number of components is greater than one (see [fit_abundances](#)).
dat	A matrix or dataframe with at least two columns, including the abundance data used to fit the multimodal gambin model and the species names. An optional third column can be provided that contains species classification data.
peak_val	A vector of of modal octave values. If peak_val = NULL, the modal octave values are taken from the model fit object.
abundances	The name of the column in dat that contains the abundance data (default = "abundance").
species	The name of the column in dat that contains the species names (default = "species").
categ	Either NULL if no species classification data are provided, or the name of the column in dat that contains the species classification data.
plot_modes	A logical argument specifying whether a barplot of the model fit with highlighted octaves should be generated. If categ = NULL a barplot is produced whereby just the modal octaves are highlighted in red. If categ is provided a barplot is produced whereby the bar for each octave is split into n parts, where n equals the number of species categories.
col.statu	A vector of colours (of length n) for the split barplot, where n equals the number of species categories.
plot_legend	Should the barplot include a legend. Only applicable when plot_modes = TRUE and categ is not NULL.
legend_location	
	If plot_legend = TRUE, where should the legend be located. Should be one of "bottomright", "bottom", "bottomleft", "left", "topleft", "top", "topright" (default), "right", or "center".

Details

The function enables greater exploration of a multimodal gambin model fit. If no species classification data are available (i.e. categ = NULL) the function returns the modal octaves of the n-component

distributions and the names of the species located in each octave. If `plot_modes = TRUE` a plot is returned with the modal octaves highlighted in red. If species classification data are provided the function also returns a summary table with the number of each species category in each octave provided. The user can then use these data to run different tests to test whether, for example, the number of species in each category in the modal octaves is significantly different than expected by chance. If `plot_modes = TRUE` a split barplot is returned whereby each bar (representing an octave) is split into the n species categories.

Species classification data should be of type character (e.g. native or invasive).

Occasionally, some of the component distributions in a multimodal gambin model fit have the same modal octave; this is more common when fitting the 3-component model. When this occurs a warning is produced, but it is not a substantive issue.

Value

An object of class `deconstruct`. The object is a list with either two or three elements. If `categ = NULL`, the list has two elements: 1) 'Peak_locations', which contains the modal octave values, and 2) 'Species_per_octave', which is a list where each element contains the species names in an octave. If `categ != NULL`, the returned object has a third element: 3) 'Summary_table', which contains a dataframe (frequency table) with the numbers of each category of species in each octave.

Author(s)

Thomas J. Matthews & Francois Rigal

Examples

```
data(categ)
fits2 = fit_abundances(categ$abundances, no_of_components = 2)
#without species classification data
deconstruct_modes(fits2, dat = categ, peak_val = NULL, abundances = "abundances",
species = "species", categ = NULL, plot_modes = TRUE)
#with species classification data
deconstruct_modes(fits2, dat = categ, categ = "status", col.statu = c("green", "red", "blue"))
#manually choose modal octaves
deconstruct_modes(fits2, dat = categ, peak_val = c(0,1))
```

---

dgambin *The mixture gambin distribution*

Description

Density, distribution function, quantile function and random generation for the mixture gambin distribution.

Usage

```
dgambin(x, alpha, maxoctave, w = 1, log = FALSE)

pgambin(q, alpha, maxoctave, w = 1, lower.tail = TRUE, log.p = FALSE)

rgambin(n, alpha, maxoctave, w = 1)

qgambin(p, alpha, maxoctave, w = 1, lower.tail = TRUE, log.p = FALSE)

gambin_exp(alpha, maxoctave, w = 1, total_species)
```

Arguments

x	vector of (non-negative integer) quantiles.
alpha	The shape parameter of the GamBin distribution.
maxoctave	The scale parameter of the GamBin distribution - which octave is the highest in the empirical dataset?
w	A vector of weights. Default, a single weight. This vector must of the same length as alpha.
log	logical; If TRUE, probabilities p are given as log(p).
q	vector of quantiles.
lower.tail	logical; if TRUE (default), probabilities are P[X <= x], otherwise, P[X > x].
log.p	logical; if TRUE, probabilities p are given as log(p).
n	number of random values to return.
p	vector of probabilities.
total_species	The total number of species in the empirical dataset

Details

dgambin gives the distribution function of a mixture gambin, so all octaves sum to 1. gambin_exp multiplies this by the total number of species to give the expected GamBin distribution in units of species, for comparison with empirical data.

Value

A vector with length MaxOctave + 1 of the expected number of species in each octave

References

Matthews, T. J., Borregaard, M. K., Gillespie, C. S., Rigal, F., Ugland, K. I., Krüger, R. F., . . . Whittaker, R. J. (2019) Extension of the gambin model to multimodal species abundance distributions. Methods in Ecology and Evolution, doi:10.1111/2041-210X.13122

Matthews, T.J., Borregaard, M.K., Ugland, K.I., Borges, P.A.V, Rigal, F., Cardoso, P. and Whittaker, R.J. (2014) The gambin model provides a superior fit to species abundance distributions with a single free parameter: evidence, implementation and interpretation. Ecography 37: 1002-1011.

Examples

```
## maxoctave is 4. So zero for x = 5
dgambin(0:5, 1, 4)

## Equal weightings between components
dgambin(0:5, alpha = c(1,2), maxoctave = c(4, 4))

## Zero weight on the second component, i.e. a 1 component model
dgambin(0:5, alpha = c(1,2), maxoctave = c(4, 4), w = c(1, 0))
expected = gambin_exp(4, 13, total_species = 200)
plot(expected, type = "l")

##draw random values from a gambin distribution
x = rgambin(1e6, alpha = 2, maxoctave = 7)
x = table(x)
freq = as.vector(x)
values = as.numeric(as.character(names(x)))
abundances = data.frame(octave=values, species = freq)
fit_abundances(abundances, no_of_components = 1)
```

---

fit_abundances	*Fit a unimodal or multimodal gambin model to a species abundance distribution*

Description

Uses maximum likelihood methods to fit the GamBin model (with a given number of modes) to binned species abundances. To control for the effect of sample size, the abundances may be subsampled prior to fitting.

Usage

```
fit_abundances(abundances, subsample = 0, no_of_components = 1, cores = 1)

fitGambin(abundances, subsample = 0)
```

Arguments

abundances	Either a vector of abundances of all species in the sample/community; or the result of `create_octaves`
subsample	The number of individuals to sample from the community before fitting the GamBin model. If subsample == 0 the entire community is used
no_of_components	
	Number of components (i.e. modes) to fit.The default (no_of_components == 1) fits the standard unimodal gambin model.
cores	No of cores to use when fitting. Use `parallel::detectCores()` to detect the number of cores on your machine.

Details

The gambin distribution is fit to the number of species in abundance octaves, as specified by the `create_octaves` function. Because the shape of species abundance distributions depend on sample size, abundances of different communities should be compared on equally large samples. The sample size can be set by the `subsample` parameter. To estimate `alpha` from a standardised sample, the function must be run several times; see the examples. The `no_of_components` parameter enables multimodal gambin distributions to be fitted. For example, setting `no_of_components` equal to 2, the bimodal gambin model is fitted. When a multimodal gambin model is fitted (with g modes), the return values are the alpha parameters of the g different component distributions, the max octave values for the g component distributions (as the max octave values for the g-1 component distributions are allowed to vary), and the and the weight parameter(s) which denote the fraction of objects within each g component distribution. When fitting multimodal gambin models (particularly on large datasets), the optimisation algorithm can be slow. In such cases, the process can be speeded up by using the `cores` parameter to enable parallel computing.

The `plot` method creates a barplot showing the observed number of species in octaves, with the fitted GamBin distribution shown as black dots. The `summary.gambin` method provides additional useful information such as confidence intervals around the model parameter estimates.

Value

The `fit_abundances` function returns an object of class `gambin`, with the `alpha`, `w` and `MaxOctave` parameters of the gambin mixture distribution, the likelihood of the fit, and the empirical distribution over octaves.

Examples

```
data(moths)
fit = fit_abundances(moths)
barplot(fit)
lines(fit, col=2)
summary(fit)
# gambin parameters based on a standardized sample size of 1000 individuals
stand_fit <- replicate(20, fit_abundances(moths, 1000)$alpha) #may take a while on slower computers
print(c(mean = mean(stand_fit), sd = sd(stand_fit)))
# a bimodal gambin model
biMod <- fit_abundances(moths, no_of_components = 2)
```

---

fly   *Brazilian Horse Fly Data*

Description

Horse flies captured using various sampling methods at different sites across Brazil.

Format

A list with two elements. The first element contains a numerical vector with the abundance of 164 fly species sampled at various sites across Brazil. The second element contains a numerical vector with the abundance of 58 fly specie sampled at a single site within Brazil using just canopy traps.

Source

This package.

Examples

```
data(fly, package = "gambin")
```

---

moths   *Williams' Rothamsted moth data*

Description

Macro-Lepidoptera captured in a light trap at Rothamsted Experimental Station during 1935.

Format

A numerical vector with the abundance of 195 moth species.

Source

Williams, C.B. (1964) Patterns in the balance of nature. Academic Press, London.

Examples

```
data(moths, package = "gambin")
```

---

mult_abundances   *Fit a unimodal gambin model to multiple species abundance distributions*

Description

Fits the unimodal gambin model to the SADs from multiple sites and returns the standardised and unstandardised alpha values.

Usage

```
mult_abundances(mult, N = 100, subsample = NULL, r = 3)
```

Arguments

`mult`	Either a matrix, dataframe or list containing the species abundance data of a set of sites. In the case of a matrix or dataframe, a given column contains the abundance data for a given site (i.e. columns are sites and rows are species; each cell is the abundance of a given species in a given site). In the case of a list, each element in the list contains the abundance data (i.e. a vector of abundances) for a given site.
`N`	The number of times to subsample the abundance data in each site to calculate mean standardised alpha.
`subsample`	The number of individuals to sample from each site before fitting the gambin model. The default is subsample = NULL, in which case subsample is set to equal the number of individuals in the site with the fewest individuals.
`r`	The number of decimal points to round the returned alpha values to (default is r = 3)

Details

Because the alpha parameter of the gambin model is dependent on sample size, when comparing the alpha values between sites it can be useful to first standardise the number of individuals in all sites. By default, the `mult_abundances` function calculates the total number of individuals in each site and selects the minimum value for standardising. This minimum number of individuals is then sampled from each site and the gambin model fitted to this subsample using `fit_abundances`, and the alpha value stored. This process is then repeated `N` times and the mean alpha value is calculated for each site. The number of individuals to be subsampled can be manually set using the `subsample` argument. The function returns a list, in which the first two elements are the mean standardised alpha values for each site, and the raw unstandardized alpha values for each site, respectively. The full set of `N` alpha values and X2 P-values for each site are also returned. As an input, the SAD data can be in the form of a matrix or dataframe, or a list. A matrix/dataframe is only for when each site (column) has abundance data for the same set of species (rows). For example, an abundance matrix of bird species on a set of islands in an archipelago. A list should be used in cases where the number of species varies between sites; for example, when comparing the SADs of samples from different countries. In this case, each element of the list contains an abundance vector from a given site.

At present, the `mult_abundances` function only fits the unimodal gambin model.

Examples

```
#simulate a matrix containing the SAD data for 10 sites (50 sp. in each)
mult <- matrix(0, nrow = 50, ncol = 10)
mult <- apply(mult, 2, function(x) ceiling(rlnorm(length(x), 0, 2.5)))

#run the mult_abundances function and view the alpha values
mm <- mult_abundances(mult, N = 10, subsample = NULL)
mm[1:2]
plot(mm$Mean.Stan.Alpha, mm$Unstan.Alpha)

#simulate a list containing the SAD of 5 sites (with varying numbers of sp.)
mult2 <- vector("list", length = 5)
```

```
for (i in 1:ncol(mult)){
 dum <- sample(mult[, i], replace = TRUE)
 rm <- round(runif(1, 0, 5), 0)
 if (rm > 0){
   rm2 <- sample(1:length(dum), rm, replace = FALSE)
   dum <- dum[-rm2]
 }
 mult2[[i]] <- dum
}

#run the mult_abundances function on the list
mm2 <- mult_abundances(mult2, N = 5, subsample = NULL)
mm2[1:2]
```

---

summary.gambin   *Summarising the results of a gambin model fit*

Description

S3 method for class 'gambin'. summary.gambin creates summary statistics for objects of class 'gambin'.The summary method generates more useful information (e.g. confidence intervals) for the user than the standard model fitting function. Another S3 method (print.summary.gambin; not documented) is used to print the output.

Usage

```
## S3 method for class 'gambin'
summary(object, confint = FALSE, n = 50, ...)
```

Arguments

object   A gambin model fit object from fit_abundances

confint   A logical argument specifying whether confidence intervals should be calcu-
                lated (via bootstrapping) for the parameters of gambin models with more than 1
                component (confidence intervals for 1 component gambin models are calculated
                automatically)

n   The number of bootstrap samples to use in generating the confidence intervals
                (for multimodal gambin models)

...   Further arguments to pass

Details

For the one-component gambin model the confidence interval for the alpha parameter is calculated automatically using an analytical solution.

For gambin models with more than one component no analytical solution for deriving the confidence intervals is known. Instead, a bootstrapping procedure can be used (using the confint and n arguments) to generate confidence intervals around the alpha and max octave parameters. However,

the process can be time-consuming, particularly for gambin models with more than two compo-
nents. Thus, the default is that confidence intervals are not automatically calculated for gambin
models with more than one component (i.e. `confint` == FALSE).

In addition, it should be noted that in certain case the confidence intervals around the alpha pa-
rameters in multi-component gambin models can be quite wide. This is due to changes in the max
octaves of the component distributions in the bootstrapped samples. It can be useful to make a plot
(e.g. a dependency boxplot) of the n alpha values against the max octave values.

Value

A list of class 'summary.gambin' with nine elements, containing useful information about the model
fit.

Examples

```
 ## Not run:
data(moths)
fit = fit_abundances(moths)
summary(fit)
# multimodal gambin models with confidence intervals
biMod <- fit_abundances(moths, no_of_components = 2)
summary(biMod, confint = TRUE, n = 5) #large n takes a long time to run

## End(Not run)
```

Index