Package: SADISA (via r-universe)

September 13, 2024

Type Package

Title Species Abundance Distributions with Independent-Species Assumption

Version 1.2

Author Rampal S. Etienne & Bart Haegeman

Maintainer Rampal S. Etienne <r.s.etienne@rug.nl>

Description Computes the probability of a set of species abundances of a single or multiple samples of individuals with one or more guilds under a mainland-island model. One must specify the mainland (metacommunity) model and the island (local) community model. It assumes that species fluctuate independently. The package also contains functions to simulate under this model. See Haegeman, B. & R.S. Etienne (2017). A general sampling formula for community structure data. Methods in Ecology & Evolution 8: 1506-1519 <doi:10.1111/2041-210X.12807>.

License GPL-3

LazyData FALSE

RoxygenNote 7.1.1

Encoding UTF-8

Depends R (>= 3.5)

Imports pracma, DDD (>= 4.1)

Suggests testthat, knitr, rmarkdown,

VignetteBuilder knitr

NeedsCompilation no

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

RemoteUrl https://github.com/rsetienne/sadisa

RemoteRef HEAD

RemoteSha f220249e82ea08ec8d716692795e5ea0ce62d8d4

Contents

`convert_fa2sf` *Converts different formats to represent multiple sample data*

Description

Converts the full abundance matrix into species frequencies If S is the number of species and M is the number of samples, then fa is the full abundance matrix of dimension S by M. The for example fa = [0 1 0;3 2 1;0 1 0] leads to sf = [0 1 0 2;3 2 1 1];

Usage

```
convert_fa2sf(fa)
```

Arguments

`fa`	the full abundance matrix with species in rows and samples in columns

Value

the sample frequency matrix

References

Haegeman, B. & R.S. Etienne (2017). A general sampling formula for community structure data. Methods in Ecology & Evolution. In press.

`datasets` *Data sets of various tropical forest communities*

Description

Various tree commnunity abundance data sets to test and illustrate the Independent Species approach.

- dset1.abunvec contains a list of 6 samples of tree abundances from 6 tropical forest plots (BCI, Korup, Pasoh, Sinharaja, Yasuni, Lambir).
- dset2.abunvec contains a list of 11 lists with one of 11 samples from BCI combined with samples from Cocoli and Sherman.
- dset3.abunvec contains a list of 6 lists with 2 samples, each from one dispersal guild, for 6 tropical forest communities (BCI, Korup, Pasoh, Sinharaja, Yasuni, Lambir).
- dset4a.abunvec contains a list of 6 samples from 6 censuses of BCI (1982, 1985, 1990, 1995, 200, 2005) with dbh > 1 cm.
- dset4b.abunvec contains a list of 6 samples from 6 censuses of BCI (1982, 1985, 1990, 1995, 200, 2005) with dbh > 10 cm.

Usage

```
data(datasets)
```

Format

A list of 5 data sets. See description for information on each of these data sets.

Author(s)

Rampal S. Etienne & Bart Haegeman

Source

Condit et al. (2002). Beta-diversity in tropical forest trees. Science 295: 666-669. See also 11. Janzen, T., B. Haegeman & R.S. Etienne (2015). A sampling formula for ecological communities with multiple dispersal syndromes. Journal of Theoretical Biology 387, 258-261.

`fitresults` *Maximum likelihood estimates and corresponding likelihood values for various fits to various tropical forest communities*

Description

Maximum likelihood estimates and corresponding likelihood values for various fits to various tropical forest communities, to test and illustrate the Independent Species approach.

- fit1a.llikopt contains maximum likelihood values of fit of pm-dl model to dset1.abunvec
- fit1a.parsopt contains maximum likelihood parameter estimates of fit of pm-dl model to dset1.abunvec
- fit1b.llikopt contains maximum likelihood values of fit of pmc-dl model to dset1.abunvec
- fit1b.parsopt contains maximum likelihood parameter estimates of fit of pmc-dl model to dset1.abunvec
- fit2.llikopt contains maximum likelihood values of fit of rf-dl model to dset1.abunvec
- fit2.parsopt contains maximum likelihood parameter estimates of fit of rf-dl model to dset1.abunvec
- fit3.llikopt contains maximum likelihood values of fit of dd-dl model to dset1.abunvec
- fit3.parsopt contains maximum likelihood parameter estimates of fit of dd-dl model to dset1.abunvec
- fit4.llikopt contains maximum likelihood values of fit of pm-dl model to dset2.abunvec (multiple samples)
- fit4.parsopt contains maximum likelihood parameter estimates of fit of pm-dl model to dset1.abunvec (multiple samples)
- fit5.llikopt contains maximum likelihood values of fit of pm-dl model to dset3.abunvec (multiple guilds)
- fit5.parsopt contains maximum likelihood parameter estimates of fit of pm-dl model to dset3.abunvec (multiple guilds)
- fit6.llikopt contains maximum likelihood values of fit of pr-dl model to dset1.abunvec
- fit6.parsopt contains maximum likelihood parameter estimates of fit of pr-dl model to dset1.abunvec
- fit7.llikopt contains maximum likelihood values of fit of pm-dd model to dset1.abunvec
- fit7.parsopt contains maximum likelihood parameter estimates of fit of pm-dd model to dset1.abunvec
- fit8a.llikopt contains maximum likelihood values of fit of pm-dd model to dset4a.abunvec
- fit8a.parsopt contains maximum likelihood parameter estimates of fit of pm-dd model to dset4a.abunvec
- fit8b.llikopt contains maximum likelihood values of fit of pm-dd model to dset4b.abunvec
- fit8b.parsopt contains maximum likelihood parameter estimates of fit of pm-dd model to dset4b.abunvec

Usage

```
data(fitresults)
```

Format

A list of 20 lists, each containing either likelihood values or the corresponding parameter estimates. See description.

Author(s)

Rampal S. Etienne & Bart Haegeman

Source

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

`integral_peak` *Computes integral of a very peaked function*

Description

computes the logarithm of the integral of exp(logfun) from 0 to Inf under the following assumptions:

Usage

```
integral_peak(
  logfun,
  xx = seq(-100, 10, 2),
  xcutoff = 2,
  ycutoff = 40,
  ymaxthreshold = 1e-12
)
```

Arguments

`logfun`	the logarithm of the function to integrate
`xx`	the initial set of points on which to evaluate the function
`xcutoff`	when the maximum has been found among the xx, this parameter sets the width of the interval to find the maximum in
`ycutoff`	set the threshold below which (on a log scale) the function is deemed negligible, i.e. that it does not contribute to the integral)
`ymaxthreshold`	sets the deviation allowed in finding the maximum among the xx

Value

the result of the integration

References

Haegeman, B. & R.S. Etienne (2017). A general sampling formula for community structure data. Methods in Ecology & Evolution. In press.

SADISA_loglik *Computes loglikelihood for requested model*

Description

Computes loglikelihood for requested model using independent-species approach

Usage

```
SADISA_loglik(abund, pars, model, mult = "single")
```

Arguments

abund	abundance vector or a list of abundance vectors. When a list is provided and mult = 'mg' (the default), it is assumed that the different vectors apply to different guilds. When mult = 'ms' then the different vectors apply to multiple samples from the same metacommunity. In this case the vectors should have equal lengths and may contain zeros because there may be species that occur in multiple samples and species that do not occur in some of the samples. When mult= 'both', abund should be a list of lists, each list representing multiple guilds within a sample
pars	a vector of model parameters or a list of vectors of model parameters. When a list is provided and mult = 'mg' (the default), it is assumed that the different vectors apply to different guilds. Otherwise, it is assumed that they apply to multiple samples.
model	the chosen combination of metacommunity model and local community model as a vector, e.g. c('pm','dl') for a model with point mutation in the metacommunity and dispersal limitation. The choices for the metacommunity model are: 'pm' (point mutation), 'rf' (random fission), 'pr' (protracted speciation), 'dd' (density-dependence). The choices for the local community model are: 'dl' (dispersal limitation), 'dd' (density-dependence).
mult	When set to 'single' (the default), the loglikelihood for a single sample is computed When set to 'mg' the loglikelihood for multiple guilds is computed. When set to 'ms' the loglikelihood for multiple samples from the same metacommunity is computed. When set to 'both' the loglikelihood for multiple guilds within multiple samples is computed.

Details

Not all combinations of metacommunity model and local community model have been implemented yet. because this requires checking for numerical stability of the integration. The currently available model combinations are, for a single sample, c('pm','dl'), c('pm','rf'), c('dd','dl'), c('pr','dl'), c('pm','dd'), and for multiple samples, c('pm','dl').

Value

loglikelihood

References

Haegeman, B. & R.S. Etienne (2017). A general sampling formula for community structure data. Methods in Ecology & Evolution 8: 1506-1519. doi: 10.1111/2041-210X.12807

Examples

```
data(datasets);
abund_bci <- datasets$dset1.abunvec[[1]];
data(fitresults);
data.paropt <- fitresults$fit1a.parsopt[[1]];
result <- SADISA_loglik(abund = abund_bci,pars = data.paropt,model = c('pm','dl'));
cat('The difference between result and the value in fitresults.RData is:',
result - fitresults$fit1a.llikopt[[1]]);
```

SADISA_ML *Performs maximum likelihood parameter estimation for requested model*

Description

Computes maximum loglikelihood and corresponding parameters for the requested model using the independent-species approach. For optimization it uses various auxiliary functions in the DDD package.

Usage

```
SADISA_ML(
  abund,
  initpars,
  idpars,
  labelpars,
  model = c("pm", "dl"),
  mult = "single",
  tol = c(1e-06, 1e-06, 1e-06),
  maxiter = min(1000 * round((1.25)^sum(idpars)), 1e+05),
  optimmethod = "subplex",
  num_cycles = 1
)
```

Arguments

abund	abundance vector or a list of abundance vectors. When a list is provided and mult = 'mg' (the default), it is assumed that the different vectors apply to different guilds. When mult = 'ms' then the different vectors apply to multiple samples. from the same metacommunity. In this case the vectors should have equal lengths and may contain zeros because there may be species that occur in multiple samples and species that do not occur in some of the samples.

initpars	a vector of initial values of the parameters to be optimized and fixed. See labelpars for more explanation.
idpars	a vector stating whether the parameters in initpars should be optimized (1) or remain fixed (0).
labelpars	a vector, a list of vectors or a list of lists of vectors indicating the labels integers (starting at 1) of the parameters to be optimized and fixed. These integers correspond to the position in initpars and idpars. The order of the labels in the vector/list is first the metacommunity parameters (theta, and phi (for protracted speciation) or alpha (for density-dependence or abundance-dependent speciation)), then the dispersal parameters (I). See the example and the vignette for more explanation.
model	the chosen combination of metacommunity model and local community model as a vector, e.g. c('pm','dl') for a model with point mutation in the metacommunity and dispersal limitation. The choices for the metacommunity model are: 'pm' (point mutation), 'rf' (random fission), 'pr' (protracted speciation), 'dd' (density-dependence). The choices for the local community model are: 'dl' (dispersal limitation), 'dd' (density-dependence).
mult	When set to 'single' (the default), the loglikelihood for a single sample and single guild is computed. When set to 'mg', the loglikelihood for multiple guilds is computed. When set to 'ms' the loglikelihood for multiple samples from the same metacommunity is computed.
tol	a vector containing three numbers for the relative tolerance in the parameters, the relative tolerance in the function, and the absolute tolerance in the parameters.
maxiter	sets the maximum number of iterations
optimmethod	sets the optimization method to be used, either subplex (default) or an alternative implementation of simplex.
num_cycles	the number of cycles of opimization. If set at Inf, it will do as many cycles as needed to meet the tolerance set for the target function.

Details

Not all combinations of metacommunity model and local community model have been implemented yet. because this requires checking for numerical stability of the integration. The currently available model combinations are, for a single sample, c('pm','dl'), c('pm','rf'), c('dd','dl'), c('pr','dl'), c('pm','dd'), and for multiple samples, c('pm','dl').

References

Haegeman, B. & R.S. Etienne (2017). A general sampling formula for community structure data. Methods in Ecology & Evolution 8: 1506-1519. doi: 10.1111/2041-210X.12807

Examples

```
utils::data(datasets);
utils::data(fitresults);
result <- SADISA_ML(
   abund = datasets$dset1.abunvec[[1]],
```

```
   initpars = fitresults$fit1a.parsopt[[1]],
   idpars = c(1,1),
   labelpars = c(1,2),
   model = c('pm','dl'),
   tol = c(1E-1, 1E-1, 1E-1)
   );
# Note that tolerances should be set much lower than 1E-1 to get the best results.
```

SADISA_sim *Simulates species abundance data*

Description

Simulates species abundance data using the independent-species approach

Usage

```
SADISA_sim(parsmc, ii, jj, model = c("pm", "dl"), mult = "single", nsim = 1)
```

Arguments

parsmc	The model parameters. For the point mutation (pm) model this is theta and I. For the protracted model (pr) this is theta, phi and I. For the density-dependent model (dd) - which can also be interpreted as the per-species speciation model, this is theta and alpha.
ii	The I parameter. When I is a vector, it is assumed that each value describes a sample or a guild depending on whether mult == 'ms' or mult == 'mg'. When mult = 'both', a list of lists must be specified, with each list element relates to a sample and contains a list of values across guilds.
jj	the sample sizes for each sample and each guild. Must have the same structure as ii
model	the chosen combination of metacommunity model and local community model as a vector, e.g. c('pm','dl') for a model with point mutation in the metacommunity and dispersal limitation. The choices for the metacommunity model are: 'pm' (point mutation), 'rf' (random fission), 'pr' (protracted speciation), 'dd' (density-dependence). The choices for the local community model are: 'dl' (dispersal limitation), 'dd' (density-dependence).
mult	When set to 'single', the loglikelihood of a single abundance vector will be computed When set to 'mg' the loglikelihood for multiple guilds is computed. When set to 'ms' the loglikelihood for multiple samples from the same metacommunity is computed. When set to 'both' the loglikelihood for multiple guilds within multiple samples is computed.
nsim	Number of simulations to perform

Details

Not all combinations of metacommunity model and local community model have been implemented yet. because this requires checking for numerical stability of the integration. The currently available model combinations are c('pm','dl').

Value

abund abundance vector, a list of abundance vectors, or a list of lists of abundance vectors, or a list of lists of lists of abundance vectors The first layer of the lists corresponds to different simulations When mult = 'mg', each list contains a list of abundance vectors for different guilds. When mult = 'ms', each list contains a list of abundance vectors for different samples from the same metacommunity. In this case the vectors should have equal lengths and may contain zeros because there may be species that occur in multiple samples and species that do not occur in some of the samples. When mult = 'both', each list will be a list of lists of multiple guilds within a sample

References

Haegeman, B. & R.S. Etienne (2017). A general sampling formula for community structure data. Methods in Ecology & Evolution 8: 1506-1519. doi: 10.1111/2041-210X.12807

`SADISA_test` *Tests SADISA for data sets included in the paper by Haegeman & Etienne*

Description

Tests SADISA for data sets included in the paper by Haegeman & Etienne

Usage

```
SADISA_test(tol = 0.001)
```

Arguments

`tol`	tolerance of the test

References

Haegeman, B. & R.S. Etienne (2017). A general sampling formula for community structure data. Methods in Ecology & Evolution. In press.

Index