Package 'comclim'

January 22, 2024

Type Package

Title Community Climate Statistics

Version 0.9.6

Date 2024-01-22

Author Benjamin Blonder

Maintainer Benjamin Blonder <benjamin.blonder@berkeley.edu>

Description Computes community climate statistics for volume and mismatch using species' climate niches either unscaled or scaled relative to a regional species pool. These statistics can be used to describe biogeographic patterns and infer community assembly processes. Includes a vignette outlining usage.

License GPL-3

Depends methods, R (>= 3.0.0)

NeedsCompilation no

Repository CRAN

Date/Publication 2024-01-22 20:42:54 UTC

R topics documented:

---

comclim-package   *Community climate analysis*

Description

Computes community climate statistics for volume and mismatch using species' climate niches either unscaled or scaled relative to a regional species pool. These statistics can be used to describe biogeographic patterns and infer community assembly processes.

Author(s)

Benjamin Blonder <bblonder@gmail.com>

References

Blonder, B., Nogues-Bravo, D., et al. Linking environmental filtering and disequilibrium to biogeography with a community climate framework *Ecology* (2014).

---

climatedeviations   *Reports community climate deviations.*

Description

Helper function to report community climate deviations. Deviations are calculated as standard effect sizes using a robust estimator as (x - quantile(null, 0.5)) / (quantile(null,0.75) - quantile(null,0.25)). Two-tailed p-values are also calculated by recentering the observed and null values to the null distribution mean, then determining the probability that the recentered observed value is either less than the null, or greater than the null.

Usage

```
climatedeviations(object)
```

Arguments

object   A CommunityClimateStatistics object.

Value

A named list of deviations, with each element containing a standard effect size (ses) and two-tailed p-value (pvalue) for each deviation

deviation_volumeMagnitude

          A two element vector for climate volume (delta)

deviation_mismatchMagnitude

          A two element vector for climate mismatch (lambda)

deviation_mismatchDirections

>A two column data frame for climate mismatch along each climate axis (lambda_i), with row names corresponding to each axis

---

climatestatistics   *Reports observed climate statistics*

Description

Helper function to report community climate statistics for the real community, i.e. not corrected for any null model.

Usage

```
climatestatistics(object)
```

Arguments

object   A CommunityClimateStatistics object

Value

A named list:

meanNiches   A dataframe with columns corresponding to climate axes, and one row for each species in the community. Each observation is the mean climate niche of that species at tinf.

inferredClimate

>A vector with names corresponding to climate axes, representing the inferred climate determined by overlapping niches of all species in the community at tinf.

observedClimate

>A vector with names corresponding to climate axes, representing the observed climate at tobs.

volumeMagnitude

>A number representing the climate volume at tinf (Delta)

mismatchMagnitude

>A number representing the magnitude of the climate mismatch at tinf relative to the climate at tobs (Lambda)

mismatchDirections

>A vector with names corresponding to climate axes, representing the climate mismatch projected along each axis at tinf relative to the climate at tobs.

---

communityclimate   *Community climate analysis*

Description

Performs a community climate analysis. The algorithm first computes the inferred climate at tinf based on sampling from the climate niches of the species in a community at tinf relative to the observed climate at tobs, and determines their volume at tinf (Delta) and mismatch at tinf relative to tobs (Lambda). The algorithm then repeats the process for a large set of null communities, performing a richness-preserving weighted sample with replacement from the regional pool, and creates a null distribution of Delta and Lambda. Finally, community climate deviations (delta and lambda) are computed as robust standard effect sizes based on the observed and null values.

Values of delta < 0 indicate environmental filtering at tinf, while values > 0 indicate environmental permissiveness at tinf and values = 0 indicate no difference between regional and local processes at tinf. Values of lambda < 0 indicate environmental equilibrium at tinf relative to climate at tobs, while values > 0 indicate environmental disequilibrium and values = 0 indicate no difference between regional and local processes for species at tinf relative to tobs.

Actual values of Delta and Lambda indicate the absolute levels of climate space occupancy (Delta) at tinf, or absolute mismatch between community composition at tinf and local climate at tobs (Lambda).

The function assumes that climate axes are on comparable (rescaled) axes, but does not test for this. You are responsible for inputting appropriate data.

Usage

```
communityclimate(object, climateaxes = NULL,
  numreplicates = 100, numsamplesperspecies = 100, verbose = TRUE)
```

Arguments

object	A `CommunityClimateInput-class` object describing the local species composition at tinf, the regional pool (and sampling weights, if provided) at tinf, the climate niches of all species at tinf, and the observed climate at tobs.
climateaxes	A named vector with the climate axes to be considered, e.g. a subset of all of the axes.
numreplicates	Number of null models to run. Larger values give more accurate p-values and effect sizes.
numsamplesperspecies	
	Number of random samples from the climate niche of each species for each pass of the algorithm. Larger values give more accurate results.
verbose	A logical flag. If true, prints output to track progress of the algorithm.

Value

A `CommunityClimateStatistics` object.

References

For more information, see the journal article described in comclim-package.

See Also

inputcommunitydata, climatedeviations, climatestatistics, CommunityClimateStatistics-class

Examples

```
# three dimensional analysis
# by construction of the example, should yield inferences of
# delta < 0 (environmental filtering) and
# lambda < 0 (environmental equilibrium)

# (uncomment next five lines - not run for CRAN speed)

# community_richness_5 <- generatedemodata()
# result_community <- communityclimate(community_richness_5,
#  climateaxes=c("ClimateAxis1","ClimateAxis2","ClimateAxis3"),
# numreplicates=25)
# summary(result_community)
```

CommunityClimateInput-class

Class "CommunityClimateInput"

Description

A class used to compactly store input for community climate analysis. Can be plotted.

Objects from the Class

Objects can be created by calls of the form new("CommunityClimateInput", ...).

Slots

species_list_tinf: Object of class "character". The names of species in the local community at tinf.

regional_pool_tinf: Object of class "character". The names of species in the the regional pool for the local community at tinf.

regional_pool_weights_tinf: Object of class "numeric" A numeric vector with same length as regional_pool_tinf, corresponding to sampling weights at tinf. Does not need to sum to one; will be normalized. If left empty, will be auto-filled with all 1s (i.e. uniform sampling).

climate_niches_tinf: Object of class "data.frame". A dataframe with climate niche informa-
tion for all species in the local community and regional pool at tinf. Should have a taxon
column, and then any number of other columns corresponding to climate axes. Each row
represents an observation of the species with a given taxon name. Usually you will fill this
dataframe by mapping geographic occurrences of species into climate space.

observed_climate_tobs: Object of class "numeric" A named numeric vector with names cor-
responding to the same climate axes as in climate_niches_tinf. Represents the observed
climate at tobs.

See Also

plot.CommunityClimateInput

Examples

```
community_richness_5 <- generatedemodata()

summary(community_richness_5)
```

CommunityClimateStatistics-class
                        Class "CommunityClimateStatistics"

Description

A class used to compactly store output of community climate analysis. Can be summarized and
plotted.

Objects from the Class

Objects can be created by calls of the form new("CommunityClimateStatistics", ...).

Slots

obsStats: Object of class "list" A list with elements reflecting community climate statistics. See
climatestatistics for a helper function and more information.

nullStats: Object of class "list" A list of lists, corresponding to output similar to obsStats for
each null community.

deviations: Object of class "list" A lis with elements reflecting community climate deviations.
See climatedeviations for a helper function and more information.

---

generatedemodata   *Simulation data for demonstration analyses.*

Description

Creates a [CommunityClimateInput-class](#) object for usage in demos. Creates a regional pool of species with normally distributed climate niches within some region of climate space. Then chooses a subset of these species with climate niches closer to the origin to represent the local community, and defines an observed climate at another location in climate space.

Usage

```
generatedemodata(num_regionalpool = 50, num_community = 5,
  num_occurrences = 40, num_climateaxes = 3, observed = 0)
```

Arguments

num_regionalpool

          Number of species in the regional pool

num_community   Number of species in the local community

num_occurrences

          Number of occurrences to simulate as the basis for each species' climate niche

num_climateaxes

          Number of climate axes

observed   The location of the observed climate. Must be a scalar; the function will copy the value to all axes.

Value

A [CommunityClimateInput-class](#) object.

Examples

```
# a five-dimensional climate space with ten species in the local community
community_demo <- generatedemodata(num_community=10, num_climateaxes=5)
```

---

inputcommunitydata   *Input data for community climate analysis*

Description

A helper function to streamline data formatting

Usage

```
inputcommunitydata(localcommunity, regionalpool,
  regionalpoolweights = numeric(), climateniches, observedclimate)
```

Arguments

localcommunity   The names of species in the local community at tinf.

regionalpool   The names of species in the the regional pool for the local community at tinf.

regionalpoolweights

A numeric vector with same length as `regional_pool_tinf`, corresponding to sampling weights at tinf. Does not need to sum to one; will be normalized. If left empty, will be auto-filled with all 1s (i.e. uniform sampling).

climateniches   A dataframe with climate niche information for all species in the local community and regional pool at tinf. Should have a `taxon` column, and then any number of other columns corresponding to climate axes. Each row represents an observation of the species with a given taxon name. Usually you will fill this dataframe by mapping geographic occurrences of species into climate space.

observedclimate

A named numeric vector with names corresponding to the same climate axes as in `climate_niches_tinf`. Represents the observed climate at tobs.

Value

A [CommunityClimateInput-class](#) object.

---

plot.CommunityClimateInput
                    Pairs plot of species and regional pool in climate space

---

Description

Produces a pairs plot representing a multidimensional climate space. The species in the community are drawn in a contrasting fashion to the species in the regional pool.

Usage

```
## S3 method for class 'CommunityClimateInput'
plot(x, climateaxes = NULL, axisnames = NULL,
  cex.community = 0.5, cex.pool = 0.25,
  pch.community = 16, pch.pool = 16,
  colors = "rainbow", ...)
```

Arguments

x	A `CommunityClimateInput` object to be plotted.
climateaxes	A vector of climate axes to be plotted, usually a subset of the axes in `object`. If empty, defaults to all axes.
axisnames	Label text for the climate axes.
cex.community	Expansion factor for the local community's species.
cex.pool	Expansion factor for the regional pool's species.
pch.community	Plotting symbol for the local community's species.
pch.pool	Plotting symbol for the regional pool's species.
colors	The name of a function used to give a color each species.
...	Other arguments to be passed to plot commands

Value

None. Used for the side-effect of producing a plot.

Examples

```
community_richness_5 <- generatedemodata()

plot(community_richness_5)
```

```
plot.CommunityClimateStatistics
```
Draws community climate diagram.

Description

Produces either a deviations plot or a community climate diagram. The deviations plot compares null distributions (black) and observed values (red) for volume, mismatch, and mismatch along each axis, along with the interquartile range of null values. The community climate diagram shows the location of the community (red points) and null communities (black points) in climate space at tinf, and compares it to the observed climate at tobs (open black point). The climate volume for each is shown as a circle, and the climate mismatch is shown as a vector connecting the inferred climate at tinf to the observed climate at tobs.

Usage

```
## S3 method for class 'CommunityClimateStatistics'
plot(x, deviations = FALSE, axisnames = NULL,
  nnull = 10, cex.axis = 0.7, cex.nullpoints = 0.3,
  cex.obspoints = 0.5, cex.names = 1.5, ...)
```

Arguments

x	A CommunityClimateStatistics object to be plotted
deviations	If TRUE, produces a deviations plot; if FALSE, produces a community climate diagram. All subsequent options are applicable only when deviations=FALSE.
axisnames	Labels to be drawn for each climate axis.
nnull	Number of null simulations to be drawn on the diagram; lower values may produce clearer plots.
cex.axis	Expansion factor for the axis ticks.
cex.nullpoints	Expansion factor for the null points.
cex.obspoints	Expansion factor for the observed points.
cex.names	Expansion factor for the climate axis labels.
...	Other arguments to be passed to plot commands

Value

None; used for the side-effect of producing a plot.

Examples

```
# (uncomment next six lines - not run for CRAN speed)

# community_richness_5 <- generatedemodata()
# result_community <- communityclimate(community_richness_5,
#  climateaxes=c("ClimateAxis1","ClimateAxis2","ClimateAxis3"),
#  numreplicates=25)

# plot(result_community,deviations=TRUE)   # deviations plot
# plot(result_community,deviations=FALSE)  # community climate diagram
```

---

summary.CommunityClimateInput

Summarizes community climate input

Description

Summarizes community climate input

Usage

```
## S3 method for class 'CommunityClimateInput'
summary(object, ...)
```

Arguments

object	The object to be summarized
...	Other arguments to be passed

```
summary.CommunityClimateStatistics
```
Summarizes community climate statistics

Description

Summarizes community climate statistics

Usage

```
## S3 method for class 'CommunityClimateStatistics'
summary(object, ...)
```

Arguments

object	The object to be summarized
...	Other arguments to be passed

Index