Package 'goeveg'

November 22, 2018

Type Package

Title Functions for Community Data and Ordinations

Version 0.4.2

Date 2018-11-22

Author Friedemann Goral [aut, cre],
Jenny Schellenberg [aut]

Maintainer Friedemann Goral <fgoral@gwdg.de>

Description A collection of functions useful in (vegetation) community analyses and ordinations. Includes automatic species selection for ordination diagrams, species response curves and rank-abundance curves as well as calculation and sorting of synoptic tables.

License GPL (>= 2)

LazyData TRUE

Depends R (>= 2.10)

Imports vegan, fields, mgcv, Hmisc, cluster

Suggests vegdata, BiodiversityR

URL http://github.com/fgoral/goeveg

BugReports http://github.com/fgoral/goeveg/issues

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2018-11-22 14:00:21 UTC

R topics documented:

---

cv *Coefficient of variation (CV)*

Description

Compute the coefficient of variation (CV). The CV, also known as relative standard deviation (RSD), is a standardized measure of dispersion of a probability distribution or frequency distribution. It is defined as the ratio of the standard deviation to the mean and is often expressed as a percentage. In contrast to the standard deviation, it enables comparison between datasets as the CV is independent of the unit in which the measurement has been taken. If `na.rm` is `TRUE` then missing values are removed before computation proceeds.

Usage

```
cv(x, na.rm = FALSE)
```

Arguments

`x`	a numeric vector
`na.rm`	logical. Should missing values be removed?

Details

The coefficient of variation (CV) should be computed only for data measured on a ratio scale, as these are the measurements that can only take non-negative values. The CV may not have any meaning for data on an interval scale.

According to *Dormann 2013* CV-values below 0.05 (5%) indicate very high precision of the data, values above 0.2 (20%) low precision. However, this is considered as a rule of thumb. In studies of highly variable systems (e.g. some ecological studies) CV values above 1 may occur.

The CV of a zero-length vector (after removal of `NA`s if `na.rm = TRUE`) is not defined and gives an error. If there is only a single value, `sd` is `NA` and `cv` returns `NA`.

References

Dormann, C. (2013). Parametrische Statistik. Verteilungen, maximum likelihood und GLM in R. *Springer*.

"What is the difference between ordinal, interval and ratio variables? Why should I care?" *GraphPad Software Inc*. http://www.graphpad.com/support/faqid/1089/.

See Also

sd

Examples

```
## Calculate CV for variable soil depth
cv(schedenenv$soil_depth)
```

---

dimcheckMDS *Stress plot/Scree plot for NMDS*

Description

This function provides a simple plot of stress values for a given number of tested dimensions (default `k = 6`) in NMDS. This stress plot (or scree plot) shows the decrease in ordination stress with an increase in the number of ordination dimensions. It is based on function `metaMDS` (`vegan` package) and uses the `monoMDS` engine.

Usage

```
dimcheckMDS(matrix, distance = "bray", k = 6, trymax = 20,
  autotransform = TRUE)
```

Arguments

`matrix`	Community data, a matrix-like object with samples in rows and species in columns.
`distance`	Dissimilarity index used in vegdist.
`k`	Number of dimensions.
`trymax`	Maximum number of random configuration for iterative search search of stable solution.
`autotransform`	Whether to use transformation (see `metaMDS`) or not. Default is `autotransform = TRUE`.

Details

Goodness of Non-metric multidimensional scaling (NMDS) is measured by stress value. The lower the stress value, the better fit of original distances/dissimilarities and projected distances in ordination diagram is reached. Stress value depends on dimensionality; it is decreasing with increasing dimensionality. On the other hand, stress-reduction does not mean to maximise interpretation capability. Low-dimensional projections are often better to interpret and are so preferable for interpretation issues. The stress plot (or sometimes also called scree plot) is a diagnostic plots to explore both, dimensionality and interpretative value. It provides dimension-dependent stress reduction and curve estimate gives indices for meaningful stress reduction with increasing dimensionality. Furthermore, another diagnostic plot for detecting best dimension for projection of NMDS, the Shepard diagram (`stressplot`) is recommended for detecting best dimensionality in NMDS.

Clarke 1993 suggests the following guidelines for acceptable stress values: <0.05 = excellent, <0.10 = good, <0.20 = usable, >0.20 = not acceptable. The plot shows the border of the 0.20 stress value limit. Solutions with higher stress values should be interpreted with caution and those with stress above 0.30 are highly suspect.

Author(s)

Jenny Schellenberg (<jschell@gwdg.de>) and Friedemann Goral (<fgoral@gwdg.de>)

References

Clarke, K. R. (1993). Non-parametric multivariate analysis of changes in community structure. *Austral J Ecol* **18:** 117-143.

See Also

metaMDS stressplot

Examples

```
## Use of function with default values
dimcheckMDS(schedenveg)

## Use of function for testing 10 dimensions
dimcheckMDS(schedenveg, k = 10)
```

---

ordiselect *Species selection for ordination plots*

Description

This function simplifies the selection of relevant species in ordination diagrams. It works with result objects from vegan package. The selection can be based upon cover abundances, frequency values and/or species fit to multivariate analysis. The result is a vector of names of the selected species and can be used for the `select` argument in ordination plots.

Usage

```
ordiselect(matrix, ord, ablim = 1, fitlim = 1, choices = c(1, 2),
  method = "axes", env, p.max = 0.05, freq = FALSE)
```

Arguments

`matrix`	Community data, a matrix-like object with samples in rows and species in columns.
`ord`	vegan ordination result object (e.g. from decorana, cca or metaMDS).
`ablim`	Proportion of species with highest abundances to be displayed. Value between 0 and 1.
`fitlim`	Proportion of species with best fit to be displayed. Value between 0 and 1.
`choices`	Axes shown.
`method`	The species fit method: `"axes"` or `"vars"`. See details for methods.
`env`	Fitted environmental variables (result object of envfit). Only used if `method = "vars"`.
`p.max`	Significance limit for variables used in `method = "vars"`.
`freq`	Whether to use cover abundances (= default) or frequencies of `matrix`. If `TRUE`, frequencies of species are used.

Details

Two methods for species fit are implemented.

- In `method = "axes"` species scores are used for selecting best fitting species. This is the default method. The basic assumption is that species that show high correlations to ordination axes have good fit to the gradient. High scores along ordination axes mean high correlation. In this method, all species with high correlations to ordination axes will be filtered.
- In `method = "vars"` environmental variables are used for selecting best fitting species. This is a distance-based approach for showing the species with best species-environment-correlation in ordination diagrams. Therefore Euclidean distances between species and environment variable centroids are calculated. Only high-responding species with very close or very far distances are considered.

If `method = "vars"` is used, the environmental variables need to be fitted with `envfit` and the result of this function must be provided to the `env` argument. The `p.max` argument allows selection of only significant variables, default is `p.max = 0.05`.

The two described methods work well both in eigenvalue-based and in distance-based ordinations. But note, that the distance-based approach for species fit is recommended for distance-based methods (e.g. NMDS), in which axes are arbitrary. If axes fit should be applied on distance-based ordination, species scores need to be calculated during the analysis, e.g. by selecting `wascores = TRUE` in `metaMDS`. On the other hand, distance calculation may be meaningless in Eigenvalue-based approaches. However, both methods provide good option of objective reduction of visible species in ordination plot for better interpretation issues.

The default for `matrix` input is a cover-abundance-matrix. This matrix should also be used for ordination.

If no limit is defined for one of the arguments `ablim, fitlim`, all species are displayed.

Author(s)

Friedemann Goral (<fgoral@gwdg.de>) and Jenny Schellenberg

Examples

```
## Calculate DCA
library(vegan)
scheden.dca <- decorana(schedenveg)

## Select the 30% most abundant species and call the result
limited <- ordiselect(schedenveg, scheden.dca, ablim = 0.3)
limited

## Use the result in plotting
plot(scheden.dca, display="n")
points(scheden.dca, display="sites")
points(scheden.dca, display="species",
   select = limited, pch=3, col="red", cex=0.7)
ordipointlabel(scheden.dca, display="species",
   select = limited, col="red", cex=0.7, add = TRUE)
```

```
## Select the 30% most frequent species with 50% best axis fit
limited <- ordiselect(schedenveg, scheden.dca, ablim = 0.3,
   fitlim = 0.5, freq = TRUE)

## Select the 30% most abundant species with 60% best environmental fit
## in NDMS for axes 1 & 3
nmds <- metaMDS(schedenveg, k = 3)   # run NMDS
env13 <- envfit(nmds, schedenenv[,2:10], choices=c(1,3))
limited13 <- ordiselect(schedenveg, nmds, ablim = 0.3, fitlim = 0.6,
   choices = c(1,3), method = "vars", env = env13)
```

---

racurve *Rank-abundance curves*

Description

This function draws a rank-abundance curve for community data. You can optionally add labels for a selected number of species. If you wish to draw multiple rank-abundance curves for selected samples use racurves.

Usage

```
racurve(matrix, main = "Rank-abundance diagram", nlab = 0,
  ylog = FALSE, frequency = FALSE)
```

Arguments

matrix	Community data, a matrix-like object with samples in rows.
main	The main title (optional).
nlab	Number of labeled species (default = 0). Species are labeled in decreasing order beginning from the highest relative abundance.
ylog	If set on TRUE the y-axis is displayed on a log-scale.
frequency	If set on TRUE frequencies of species are calculated instead of relative abundances.

Value

Returns an (invisible) list composed of:

abund	abundance of each species (in decreasing order)
rel.abund	relative abundance of each species (in decreasing order)
freq	frequency of each species (in decreasing order)

Details

Rank abundance curves or Whittaker plots (see *Whittaker 1965*) are used to display relative species abundance as biodiversity component. They are a means to visualize species richness and species evenness.

Author(s)

Friedemann Goral (<fgoral@gwdg.de>)

References

Whittaker, R. H. (1965). Dominance and Diversity in Land Plant Communities: Numerical relations of species express the importance of competition in community function and evolution. *Science* **147 :** 250-260.

See Also

racurves for multiple curves and rankabundance from package BiodiversityR for a more sophisticated function

Examples

```
## Draw rank-abundance curve
racurve(schedenveg)

## Draw rank-abundance curve and label first 5 species
racurve(schedenveg, nlab = 5)

## Draw rank-abundance curve with log-scaled axis
racurve(schedenveg, ylog = TRUE)

## Draw rank-abundance curve with frequencies and no main title
racurve(schedenveg, frequency = TRUE, nlab = 1, main = "")
```

---

racurves *Multiple rank-abundance curves*

Description

This function draws multiple rank-abundance curves for selected samples into one diagram. If you wish to draw a simple rank-abundance curve see racurve.

Usage

```
racurves(matrix, main = "Rank-abundance diagram", bw = TRUE)
```

Arguments

matrix	Community data, a matrix-like object with samples in rows and species in columns. Rank-abundance curves are drawn for all selected rows (samples).
main	The main title (optional).
bw	If set on FALSE the lines will be drawn in colors instead of black/white lines with different line types.

Details

Rank abundance curves or Whittaker plots (see *Whittaker 1965*) are used to display relative species abundance as biodiversity component. They are a means to visualize species richness and species evenness.

The axes of the diagram will be scaled according automatically. As the line type is used to differenciate between samples, a maximum of 6 curves per diagram is feasible in black/white mode.

Author(s)

Friedemann Goral (<fgoral@gwdg.de>)

References

Whittaker, R. H. (1965). Dominance and Diversity in Land Plant Communities: Numerical relations of species express the importance of competition in community function and evolution. *Science* **147 :** 250-260.

See Also

racurve for a simple curve and rankabundance from package BiodiversityR for a more sophisticated function

Examples

```
## Draw multiple rank-abundance curves for selected samples
racurves(schedenveg[c(1,7,20,25), ])

## Draw multiple rank-abundance curves for selected samples with coloured lines
racurves(schedenveg[c(1,7,20,25), ], bw = FALSE)
```

---

schedenenv *Header data for Vegetation releves from Scheden*

Description

An example vegetation dataset containing 28 grassland releves from Scheden, Niedersachsen, Germany. The releves were done May 2016 during a students field course at the University of Goettingen. Locations at the study site are based on the diploma thesis from *Eichholz (1997)*

Usage

```
schedenenv
```

Format

A data frame with 28 rows (samples) and 10 variables

- comm: Plant community as defined in 1997: *Arrhenatheretum* or *Gentiano-Koelerietum*
- altit: Altitude (m)
- north: North value as cosinus of aspect
- slope: Slope (degrees)
- cov_herb: Cover of herb layer (%)
- cov_litt: Cover of litter (%)
- cov_moss: Cover of mosses (%)
- cov_opensoil: Cover of open soil (%)
- height_herb: Average height of herb layer (cm)
- soil_depth: Soil depth (cm)

References

Eichholz, A. (1997): Wiesen und Magerrasen am Suedhang des Hohen Hagen. Diplomarbeit Biologie, University of Goettingen.

---

schedenveg *Vegetation releves from Scheden*

Description

An example vegetation dataset containing 28 grassland releves from Scheden, Niedersachsen, Germany. The releves were done May 2016 during a students field course at the University of Goettingen. Locations at the study site are based on the diploma thesis from *Eichholz (1997)*

Usage

```
schedenveg
```

Format

A data frame with 28 rows (samples) and 155 variables (species)

References

Eichholz, A. (1997): Wiesen und Magerrasen am Suedhang des Hohen Hagen. Diplomarbeit Biologie, University of Goettingen.

sem *Standard error of the mean (SEM)*

Description

Compute the standard error of the mean (SEM). The SEM is the standard deviation of the sample-mean's estimate of a population mean. It therefore describes the accuracy of the calculation of a sample's mean. If `na.rm` is `TRUE` then missing values are removed before computation proceeds.

Usage

```
sem(x, na.rm = FALSE)
```

Arguments

`x`	a numeric vector
`na.rm`	logical. Should missing values be removed?

Details

The SEM of a zero-length vector (after removal of `NA`s if `na.rm = TRUE`) is not defined and gives an error. The SEM of a length-one vector is `NA`.

See Also

`sd`

Examples

```
## Calculate mean and SEM for variable soil depth
mean(schedenenv$soil_depth)
sem(schedenenv$soil_depth)
```

specresponse *Species response curves*

Description

This function fits species response curves to visualize species responses to environmental gradients or ordination axes. It is based on Logistic Regression using Generalized Linear Models (GLMs) or Generalized Additive Models (GAMs) with integrated smoothness estimation. The function can draw response curves for single or multiple species.

Usage

```
specresponse(species, var, main, xlab, model = "auto", method = "env",
  axis = 1, points = FALSE, bw = FALSE)
```

Arguments

species	Species data (either a community matrix object with samples in rows and species in columns - response curves are drawn for all (selected) columns; or a single vector containing species abundances per plot).
var	Vector containing environmental variable (per plot) **OR** vegan ordination result object if `method = "ord"`.
main	Optional: Main title.
xlab	Optional: Label of x-axis.
model	Defining the assumed species response: Default `model = "auto"` selects the model automatically based on AIC. Other methods are `model = "linear"` (linear response), `model = "unimodal"` (unimodal response), `model = "bimodal"` (bimodal response) and `model = "gam"` (using GAM with regression smoother).
method	Method defining the type of variable. Default `method = "env"` fits a response curve to environmental variables. Alternatively `method = "ord"` fits a response along ordination axes.
axis	Ordination axis (only if `method = "ord"`).
points	If set on `TRUE` the species occurrences are shown as points. To avoid overlapping they are shown with vertical offset.
bw	If set on `TRUE` the lines will be drawn in black/white with different line types instead of colors.

Details

For response curves based on environmental gradients the argument `var` takes a single vector containing the variable corresponding to the species abundances.

For a response to ordination axis (`method = "ord"`) the argument `var` requires a `vegan` ordination result object (e.g. from `decorana`, `cca` or `metaMDS`). First axis is used as default.

By default the response curves are drawn with automatic GLM model selection based on AIC out of GLMs with 1 - 3 polynomial degrees (thus excluding bimodal responses which must be manually defined). The GAM model is more flexible and choses automatically between an upper limit of 3 - 6 degrees of freedom for the regression smoother.

Available information about species is reduced to presence-absence as species abundances can contain much noise (being affected by complex factors) and the results of Logistic Regression are easier to interpret showing the "probabilities of occurrence". Be aware that response curves are only a simplification of reality (model) and their shape is strongly dependent on the available dataset.

Author(s)

Friedemann Goral (<fgoral@gwdg.de>)

Examples

```
## Draw species response curve for one species on environmental variable
## with points of occurrences
specresponse(schedenveg$ArrElat, schedenenv$soil_depth, points = TRUE)
```

```
## Draw species response curve on environmental variable with custom labels
specresponse(schedenveg$ArrElat, schedenenv$soil_depth, points = TRUE,
       main = "Arrhenatherum elatius", xlab = "Soil depth")

## Draw species response curve on ordination axes
## First calculate DCA
library(vegan)
scheden.dca <- decorana(schedenveg)

# Using a linear model on first axis
specresponse(schedenveg$ArrElat, scheden.dca, method = "ord", model = "linear")
# Using an unimodal model on second axis
specresponse(schedenveg$ArrElat, scheden.dca, method = "ord", axis = 2, model = "unimodal")

## Community data: species (columns) need to be selected; call names() to get column numbers
names(schedenveg)
## Draw multiple species response curves on variable in black/white
specresponse(schedenveg[ ,c(9,18,14,19)], schedenenv$height_herb, bw = TRUE)

## Draw the same curves based on GAM
specresponse(schedenveg[ ,c(9,18,14,19)], schedenenv$height_herb, bw = TRUE, model = "gam")

## Draw multiple species response curves on variable with
## custom x-axis label and points of occurrences
specresponse(schedenveg[ ,c(9,18,14,19)], schedenenv$height_herb,
    xlab = "Height of herb layer (cm)", points = TRUE)

## Draw multiple species response curves on ordination axes
specresponse(schedenveg[ ,c(9,18,14,19)], scheden.dca, method = "ord")
specresponse(schedenveg[ ,c(9,18,14,19)], scheden.dca, method = "ord", axis = 2)
```

`synsort` *Sorting functions for synoptic tables*

Description

Synoptic tables are a tool for interpretation of cluster species composition. This function provides sorting options for synoptic tables, sorting criteria can be either values in synoptic tables, such as frequencies, as well as combined criteria with considering differential character, too. Sorting algorithm aims to sort species in given cluster column order to blocked structure. Thereby, species with high frequencies and/or differential character are displayed blocked for each cluster or several neighbouring clusters.

Usage

```
synsort(syn1, syn2 = syn1, cluster, method = "allspec", min1 = 0,
  min2 = 0, relate2 = "entire")
```

Arguments

syn1	Input synoptic table 1 (as dataframe) with priority entries for sorting. Usually dataframe from syntable function output, but function should work with every synoptic table input, as long as formats are appropriate. The values of this table will be displayed in the final output table.
syn2	Optional second input table with additional sorting criteria. Note that values of second input table will be considered in sorting, but not be displayed in final synoptic table with method = "allspec".
cluster	Integer vector with classification cluster identity. Ensure matching order of cluster identity and samples in dataframe for correct allocation of cluster numbers to samples.
method	Sorting algorithm (method = c("allspec", "p_diff", "n_diff", "pn_diff", "accspec", "all_d See Details.
min1	Treshold minimum value for considering species of syn1 in ordering algorithm. Species below that minimum will neither be considered in algorithm nor displayed in final synoptic table, but will be listed in the $others output.
min2	Treshold minimum value for considering species of syn2 in ordering algorithm. Species below that minimum will neither be considered in algorithm nor displayed in final synoptic table, but will be listed in the $others vector.
relate2	Specifies relation of given second table minimum values to either related to entire dataset (default) or to each cluster only (relate2 = c("entire", "cluster")).

Value

Returns a list composed of:

$output	sorting method description
$species	species sorting criteria
$samplesize	sample sizes in clusters
$syntable	sorted synoptic table
$others	species that failed to be included in the final table due to treshold values given by min1 and min2
$differential	In case of combined sorting with considering differential species character, a table with differential character of species.

Details

Six types of synoptic tables can be created with this function.

method = "allspec" creates a sorted synoptic table basing on numeric input tables, e.g. common percentage frequency tables. Sorting criteria can be either given by one input table (syn1), as well as by two input tables (syn1, syn2). Thereby, only values of syn1 will be shown in the final sorted table, but values of syn2 will be considered, too. The second minimum treshold (min2) for values in syn2 will be either applied on single clusters (e.g. recommended for phi values with specifying relate2 = "cluster") for having minimum phi/cluster or e.g. on total frequencies in entire dataset for excluding rare species from synoptic table, applying the minimum treshold on the entire dataset (relate2 = "entire").

With including differential species character as sorting criterion (`method = c("p_diff", "n_diff", "pn_diff", "accspe`
input table `syn1` must be numeric, the second one with information on differential character (output from `syntable` function with `type="diffspec"`). Again, algorithm detects highest cluster values of species in `syn1` as base for sorting, but will sort them considering differentiating character criterion (from second input table `syn2`). Species with high values in `syn1` AND differential character will then be listed on the top of a species block. Within differentiating species, prevalence of diagnostic character is considered by favoring positive and/or cluster-specific differential character. Available types are:

`method = "p_diff"` creates a synoptic table of diagnostic species with numerical values of input table `syn1`

`method = "accspec"` creates a synoptic table of non-diagnostic species with numerical values of input table `syn1`

`method = "all_diff"` showing all diagnostic and non-diagnostic species

Author(s)

Jenny Schellenberg (<jschell@gwdg.de>)

References

Bruelheide, H. (2000): A new measure of fidelity and its application to defining species groups. - *Journal of Vegetation Science* 11: 167-178.

Chytry, M., Tichy, L., Holt, J., Botta-Dukat, Z. (2002): Determination of diagnostic species with statistical fidelity measures. *Journal of Vegetation Science* 13: 79-90.

Sokal, R.R. & Rohlf, F.J. (1995): Biometry. 3rd edition Freemann, New York.

Tsiripidis, I., Bergmeier, E., Fotiadis, G. & Dimopoulos, P. (2009): A new algorithm for the determination of differential taxa. - *Journal of Vegetation Science* 20: 233-240.

See Also

`syntable`

Examples

```
## Synoptic table of Scheden vegetation data:
library(cluster)
pam1 <- pam(schedenveg, 4)

## Unordered synoptic tables
# Unordered synoptiv percentage frequency table
unordered <- syntable(schedenveg, pam1$clustering, abund = "perc",
                      type = "percfreq")
# Differential species analysis
differential <- syntable(schedenveg, pam1$clustering, abund = "perc",
                         type = "diffspec")
# Fidelity phi
phitable <- syntable(schedenveg, pam1$clustering, abund = "perc",
                     type = "phi")
```

```
## Common complete synoptic table: sort by percentage frequency,
## show all species
sorted <- synsort(syn1 = unordered$syntable, cluster = pam1$clustering,
                  method = "allspec", min1 = 0)
sorted             # view results

## Synoptic table, with only positive differentiating species with
## minimum 25% frequency in table
positive <- synsort(syn1 = unordered$syntable, syn2 = differential$syntable,
                    cluster = pam1$clustering, method = "p_diff", min1 = 25)
positive           # view results

## Synoptic table, with percentage frequency (only species >25%) and
## differential character.
complete <- synsort(syn1 = unordered$syntable, syn2 = differential$syntable,
                    cluster = pam1$clustering, method = "all_diff", min1 = 25)
complete

## Synoptic table, species with minimum phi-value of 0.3, show
## percentage frequency
phi_complete <- synsort(syn1 = unordered$syntable, syn2 = phitable$syntable,
                     cluster = pam1$clustering, method = "allspec", min1 = 25, min2 = 0.3)
phi_complete

## Synoptic table with only phi values
phi_table <- synsort(syn1 = phitable$syntable, cluster = pam1$clustering,
                     method = "allspec", min1 = 0.3)
phitable

## Synoptic table showing diagnostic phi phi-values (>=0.3) and
## differential character
phi_diff <- synsort(syn1 = phitable$syntable, syn2 = differential$syntable,
                    cluster = pam1$clustering, method = "all_diff", min1 = 0.3)
phi_diff
```

`syntable` *Synoptic tables and calculation of fidelity and differential species character*

Description

This function calculates an unordered synoptic table for vegetation unit analysis from a species-sample dataframe and a numeric vector of cluster identity input. Synoptic table options for displaying species in clusters is absolute or percentage frequency, mean/median cover values, differential character (positive, negative, positive-negative, according to Tsiripidis et al. 2009) or fidelity phi (Bruelheide 2000). Created unordered synoptic table can be sorted automatically with `synsort` function in this package.

Usage

```
syntable(spec, cluster, abund = "perc", type = "percfreq")
```

Arguments

spec	Species matrix or dataframe with species in columns and samples in rows. Values must be numeric ("." as decimal character) or integer. Missing values, NA or NaN are not allowed. Species and sample names must be defined as column- and rownames.
cluster	Integer vector with classification cluster identity. Ensure matching order of cluster identity and samples in dataframe for correct allocation of cluster numbers to samples.
abund	Data input type. Define whether input species matrix or dataframe is presence/absence data (abund = "freq") or percentage cover (abund = "perc", default).
type	Type of synoptic table output, type = c("percfreq", "totalfreq", "mean", "median", "diffspec See Details for description of options.

Value

The function returns a list of result components.

abund	abundance of each species (in decreasing order)
$syntable	unordered synoptic table for given species and clusters
$others	names vector of species that are not included in synoptic table due to failing treshold values for min1 and/or min2
$samplesize	total samples in clusters

Additionally for differential taxa calculation:

$onlydiff	synoptic table only with differentiating taxa
$differentials	lists diagnostic taxa for each cluster

Details

For synoptic table calculation, six types are available.

- type = "percfreq" Default, creates a percentage frequency table
- type = "totalfreq" Creates an absolute frequency table
- type = "mean" Calculates mean of species values per cluster
- type = "median" Calculates median of species values per cluster
- type = "diffspec" Calculates differential character of species according to Tsiripidis et al. 2009. Synoptic table gives specification of positive (p), negative (n), positive-negative (pn) or no differential character (-). Consider that differential character is always restricted to some and not necessarily all of the other units, so regarding percentage frequency is essential for correct interpretation of diagnostic character of species.

- `type = "phi"` Calculates fidelity measure phi (algorithm basing on Sokal & Rohlf 1995, Bruelheide 2000). Values are ranging between -1 and 1 with high values near 1 indicating high fidelity.

For sorting synoptic tables, use `synsort` function. See also `syntab` function from `vegdata` package for creating ordered synoptic tables from turboveg inputs.

Author(s)

Jenny Schellenberg (<jschell@gwdg.de>)

References

Bruelheide, H. (2000): A new measure of fidelity and its application to defining species groups. - *Journal of Vegetation Science* 11: 167-178.

Chytry, M., Tichy, L., Holt, J., Botta-Dukat, Z. (2002): Determination of diagnostic species with statistical fidelity measures. *Journal of Vegetation Science* 13: 79-90.

Sokal, R.R. & Rohlf, F.J. (1995): Biometry. 3rd edition Freemann, New York.

Tsiripidis, I., Bergmeier, E., Fotiadis, G. & Dimopoulos, P. (2009): A new algorithm for the determination of differential taxa. - *Journal of Vegetation Science* 20: 233-240.

See Also

`synsort`

Examples

```
## Synoptic table of Scheden vegetation data
library(cluster)
pam1 <- pam(schedenveg, 4)  # PAM clustering with 4 clusters output

## 1) unordered synoptic percentage frequency table
unordered <- syntable(schedenveg, pam1$clustering, abund = "perc",
                      type = "percfreq")
unordered                   # view results

## 2) differential species analysis
differential <- syntable(schedenveg, pam1$clustering, abund = "perc",
                         type = "diffspec")
# show complete table with differential character of species
differential$syntable
# list differential species for second cluster
differential$differentials$group2

## 3) Synoptic table with phi fidelity
phitable <- syntable(schedenveg, pam1$clustering, abund = "perc",
                     type = "phi")
phitable
```

Index