

Package ‘zetadiv’

October 14, 2022

Title Functions to Compute Compositional Turnover Using Zeta Diversity

Version 1.2.1

Date 2022-06-08

Description Functions to compute compositional turnover using zeta-diversity, the number of species shared by multiple assemblages. The package includes functions to compute zeta-diversity for a specific number of assemblages and to compute zeta-diversity for a range of numbers of assemblages. It also includes functions to explain how zeta-diversity varies with distance and with differences in environmental variables between assemblages, using generalised linear models, linear models with negative constraints, generalised additive models, shape constrained additive models, and I-splines.

Depends R (>= 4.0.0), scam

License GPL-3

Imports stats, graphics, grDevices, utils, car, mgcv, vegan, geodist, npls, glm2

LazyData true

Encoding UTF-8

RoxygenNote 7.2.0

NeedsCompilation no

Author Guillaume Latombe [aut, cre],
Melodie A. McGeoch [aut],
David A. Nipperess [aut],
Cang Hui [aut]

Maintainer Guillaume Latombe <latombe.guillaume@gmail.com>

Repository CRAN

Date/Publication 2022-06-10 22:30:02 UTC

R topics documented:

bird.env.coarse 2

bird.env.fine	3
bird.spec.coarse	4
bird.spec.fine	5
glm.cons	6
glm.fit.cons	8
Ispline	9
Marion.env	10
Marion.species	11
pie.neg	12
Plot.ispline	13
Plot.zeta.ddecay	15
Plot.zeta.ddecays	16
Plot.zeta.decline	17
Plot.zeta.scale.min.dist	18
Plot.zeta.scale.regular	20
Predict.msgdm	21
Reg.ispline	23
rescale.min.dist	25
rescale.regular	27
Return.ispline	28
Zeta.ddecay	29
Zeta.ddecays	32
Zeta.decline.ex	34
Zeta.decline.mc	37
Zeta.msgdm	40
Zeta.order.ex	44
Zeta.order.mc	46
Zeta.order.mc.mult	49
Zeta.sam.sensitivity	52
Zeta.scale.min.dist	54
Zeta.scale.regular	57
Zeta.varpart	59

Index **62**

bird.env.coarse *South-East Australia Environmental Dataset at Coarse Scale*

Description

Projected coordinates and environmental variables in 123, 100 x 100 km sites.

Usage

data(bird.env.coarse)

Format

A data frame with 123 rows (sites) and 9 columns (xy coordinates and environmental variables).

Details

The data set contains the following variables:

- x: x-position in meters in UTM 53 South projection
- y: y-position in meters in UTM 53 South projection
- Natural: Proportion of area of conservation and natural environments
- Irrigated: Proportion of area of production from irrigated agriculture and plantations
- Water: Proportion of area of water features
- Elevation: Elevation
- ApP: Area per person
- Temp: Temperature
- Precip: Precipitation

Location: Australia – 51° 27' 2.27" S, 135° 21' 35.19" E

Data owners: ABARES, Australian Bureau of Statistics, GEBCO, WorldClim

References

http://data.daff.gov.au/anrdl/metadata_files/pa_luav4g9abl07811a00.xml

<http://www.gebco.net/>

<http://www.abs.gov.au/AUSSTATS/abs@.nsf/DetailsPage/1270.0.55.0072011?>

<http://www.worldclim.org/>

Hijmans, R.J., Cameron, S.E., Parra, J.L., Jones, P.G. & Jarvis, A. (2005) Very high resolution interpolated climate surfaces for global land areas. *International journal of climatology*, 25, 1965-1978.

bird.env.fine

South-East Australia Environmental Dataset at Fine Scale

Description

Projected coordinates and environmental variables in 604, 25 x 25 km contiguous sites.

Usage

data(bird.env.fine)

Format

A data frame with 604 rows (sites) and 9 columns (xy coordinates and environmental variables).

Details

The data set contains the following variables:

- x: x-position in meters in UTM 53 South projection
- y: y-position in meters in UTM 53 South projection
- Natural: Proportion of area of conservation and natural environments
- Irrigated: Proportion of area of production from irrigated agriculture and plantations
- Water: Proportion of area of water features
- Elevation: Elevation
- ApP: Area per person
- Temp: Temperature
- Precip: Precipitation

Location: Australia – 50° 33' 5.03" S, 135° 21' 10.40" E

Data owners: ABARES, Australian Bureau of Statistics, GEBCO, WorldClim

References

http://data.daff.gov.au/anrdl/metadata_files/pa_luav4g9abl07811a00.xml

<http://www.gebco.net/>

<http://www.abs.gov.au/AUSSTATS/abs@.nsf/DetailsPage/1270.0.55.0072011?>

<http://www.worldclim.org/>

Hijmans, R.J., Cameron, S.E., Parra, J.L., Jones, P.G. & Jarvis, A. (2005) Very high resolution interpolated climate surfaces for global land areas. *International journal of climatology*, 25, 1965-1978.

bird.spec.coarse	<i>Australia Bird Atlas Species Occurrence Dataset at Coarse Scale over South-East Australia</i>
------------------	--

Description

Inventory of bird species occurrence in 123, 100 x 100 km sites.

Usage

```
data(bird.spec.coarse)
```

Format

A data frame with 123 rows (sites) and 193 columns (xy coordinates and species).

Details

- x: x-position in meters in UTM 53 South projection
- y: y-position in meters in UTM 53 South projection
- columns 3-193: bird species occurrence

The original bird occurrence data were arranged into a continuous grid covering South-East Australia. Only cells whose richness was within 10 percents of real estimated richness are included here, so that the data corresponds to presence-absence data.

Location: Australia – 51° 27' 2.27" S, 135° 21' 35.19" E

Data owner: BirdLife Australia

References

Barrett, G., Silcocks, A., Barry, S., Cunningham, R. & Poulter, R. (2003) The new atlas of Australian birds. Royal Australasian Ornithologists Union, Melbourne, 1-824.

bird.spec.fine	<i>Australia Bird Atlas Species Occurrence Dataset at Fine Scale over South-East Australia</i>
----------------	--

Description

Inventory of bird species occurrence in 604, 25 x 25 km sites.

Usage

data(bird.spec.fine)

Format

A data frame with 604 rows (sites) and 193 columns (xy coordinates and species).

Details

- x: x-position in meters in UTM 53 South projection
- y: y-position in meters in UTM 53 South projection
- columns 3-192: bird species occurrence

Location: Australia – 50° 33' 5.03" S, 135° 21' 10.40" E

Data owner: BirdLife Australia

The original bird occurrence data were arranged into a continuous grid covering South-East Australia. Only cells whose richness was within 10 percents of real estimated richness are included here, so that the data corresponds to presence-absence data.

References

Barrett, G., Silcocks, A., Barry, S., Cunningham, R. & Poulter, R. (2003) The new atlas of Australian birds. Royal Australasian Ornithologists Union, Melbourne, 1-824.

glm.cons	<i>Fitting Generalized Linear Models with constraint on the coefficients signs</i>
----------	--

Description

glm.cons is an adaptation of function glm2 from package {glm2} in which the least squares estimation is replaced by a regression with signs constraint on the coefficients using function nnnpls from package {nnls}.

Usage

```
glm.cons(  
  formula,  
  family = stats::gaussian(),  
  data,  
  weights,  
  subset,  
  na.action,  
  start = NULL,  
  etastart,  
  mustart,  
  offset,  
  control = list(...),  
  model = TRUE,  
  method = "glm.fit.cons",  
  cons = -1,  
  cons.inter = 1,  
  x = FALSE,  
  y = TRUE,  
  contrasts = NULL,  
  ...  
)
```

Arguments

formula	as for glm
family	as for glm
data	as for glm
weights	as for glm
subset	as for glm
na.action	as for glm
start	as for glm
etastart	as for glm
mustart	as for glm

offset	as for glm
control	as for glm
model	as for glm
method	the method used in fitting the model. The default method "glm.fit.cons" uses function nnnpls from package nnls instead of lm.fit to impose the sign of the coefficients. As in glm, the alternative method "model.frame" returns the model frame and does no fitting.
cons	type of constraint. Default is -1 for negative coefficients on the predictors. The other option is 1 for positive coefficients on the predictors.
cons.inter	type of constraint for the intercept. Default is 1 for positive intercept, suitable for Gaussian family. The other option is -1 for negative intercept, suitable for binomial family.
x	as for glm
y	as for glm
contrasts	as for glm
...	as for glm

Value

The value returned by `glm.cons` has exactly the same structure as the value returned by `glm` and `glm.2`.

References

Marschner, I.C. (2011) `glm2`: Fitting generalized linear models with convergence problems. *The R Journal*, 3(2), 12-15.

See Also

[glm](#), [glm2](#)

Examples

```
## Dobson (1990) Page 93: Randomized Controlled Trial :
counts <- c(18,17,15,20,10,20,25,13,12)
outcome <- gl(3,1,9)
treatment <- gl(3,3)
print(d.AD <- data.frame(treatment, outcome, counts))
glm.D93 <- glm.cons(counts ~ outcome + treatment, family = poisson())
glm.D93.ngl <- glm.cons(counts ~ outcome + treatment, family = poisson(),
  method="glm.fit.cons")
summary(glm.D93)
summary(glm.D93.ngl)
```

glm.fit.cons	<i>Generalized Linear Models fitting method with negative coefficients constraint</i>
--------------	---

Description

glm.fit.cons is an adaptation of function glm.fit2 from package {glm2} in which the least squares estimation is replaced by a non-positive regression using function nnnpls from package {nnls}.

Usage

```
glm.fit.cons(  
  x,  
  y,  
  weights = rep(1, nobs),  
  cons = -1,  
  cons.inter = 1,  
  start = NULL,  
  etastart = NULL,  
  mustart = NULL,  
  offset = rep(0, nobs),  
  family = stats::gaussian(),  
  control = list(),  
  intercept = TRUE  
)
```

Arguments

x	as for glm.fit
y	as for glm.fit
weights	as for glm.fit
cons	type of constraint. Default is -1 for negative coefficients on the predictors. The other option is 1 for positive coefficients on the predictors.
cons.inter	type of constraint for the intercept. Default is 1 for positive intercept, suitable for Gaussian family. The other option is -1 for negative intercept, suitable for binomial family.
start	as for glm.fit
etastart	as for glm.fit
mustart	as for glm.fit
offset	as for glm.fit
family	as for glm.fit
control	as for glm.fit
intercept	as for glm.fit

Value

The value returned by `glm.fit.cons` has exactly the same structure as the value returned by `glm.fit` and `glm.fit2`.

References

Marschner, I.C. (2011) `glm2`: Fitting generalized linear models with convergence problems. *The R Journal*, 3(2), 12-15.

See Also

[glm.fit](#), [glm.fit2](#)

Examples

```
## Dobson (1990) Page 93: Randomized Controlled Trial :
counts <- c(18,17,15,20,10,20,25,13,12)
outcome <- gl(3,1,9)
treatment <- gl(3,3)
print(d.AD <- data.frame(treatment, outcome, counts))
glm.D93 <- glm.cons(counts ~ outcome + treatment, family = poisson())
glm.D93.ngl <- glm.cons(counts ~ outcome + treatment, family = poisson(),
  method="glm.fit.cons")
summary(glm.D93)
summary(glm.D93.ngl)
```

Ispline

Transform data using I-splines

Description

Evaluates the I-splines for all variables of a data frame, as performed in `Zeta.msgdm`.

Usage

```
Ispline(dat, order.ispline = 2, kn.ispline = 1, rescale = 0)
```

Arguments

<code>dat</code>	A data frame whose columns are variables to be transformed using I-splines.
<code>order.ispline</code>	Order of the I-spline.
<code>kn.ispline</code>	Number of knots in the I-spline.
<code>rescale</code>	Indicates how to rescale the values between 0 and 1. Default is 0, which divides the data by the maximum value. Any other value corresponds to setting the minimum value to 0.

Value

Ispline returns a data frame with the same number of rows as `dat` and `ncol(dat) * (order.ispline + kn.ispline)` columns.

References

- Ramsay, J. O. (1988). Monotone regression splines in action. *Statistical Science*, 425-441.
- Ferrier, S., Manion, G., Elith, J., & Richardson, K. (2007). Using generalized dissimilarity modelling to analyse and predict patterns of beta diversity in regional biodiversity assessment. *Diversity and Distributions*, 13(3), 252-264.

See Also

[Zeta.msgdm](#)

Examples

```
utils::data(bird.env.coarse)
data.env <- bird.env.coarse[,3:9]
data.env.splines <- Ispline(data.env)
```

Marion.env

Marion Island Environmental Dataset

Description

Geographic coordinates, altitude and island side (East, West) at 12 plots (4 transects and 3 altitudes) on Marion Island.

Usage

```
data(Marion.env)
```

Format

A data frame with 12 rows (plots) and 4 columns (variables).

Details

The data set contains the following variables:

- x: x-position in meters in UTM 37 projection
- y: y-position in meters in UTM 37 projection
- Altitude: mean elevation
- Side: cardinal (East or West) side of the island

Location: Marion Island – 46° 53' 34.2" S, 37 degrees 45' 02.3" E

Data owner: Melodie A. McGeoch

References

Nyakatya, M.J. & McGeoch, M.A. (2008). Temperature variation across Marion Island associated with a keystone plant species (*Azorella selago* Hook. (Apiaceae)). *Polar Biology*, 31, 139-151.

McGeoch, M.A., Le Roux, P.C., Hugo, E.A. & Nyakatya, M.J. (2008). Spatial variation in the terrestrial biotic system. *The Prince Edward Islands: Land-Sea Interactions in a Changing World* (ed. by S.L. Chown and P.W. Froneman), pp. 245-276. African SunMedia, Stellenbosch.

Marion.species

Marion Island Species Presence-Absence Dataset

Description

Inventory of springtails and mite species presence-absence in 12 plots (4 transects and 3 altitudes) on Marion Island.

Usage

data(Marion.species)

Format

A data frame with 12 rows (plots) and 33 columns (species).

Details

The data set contains the following variables:

- x: x-position in meters in UTM 37 South projection
- y: y-position in meters in UTM 37 South projection
- columns 3-24: mite species presence absence
- columns 25-33: springtail species presence absence

Location: Marion Island – 46° 53' 34.2" S, 37 degrees 45' 02.3" E

Data owner: Melodie A. McGeoch

References

Nyakatya, M.J. & McGeoch, M.A. (2008). Temperature variation across Marion Island associated with a keystone plant species (*Azorella selago* Hook. (Apiaceae)). *Polar Biology*, 31, 139-151.

McGeoch, M.A., Le Roux, P.C., Hugo, E.A. & Nyakatya, M.J. (2008). Spatial variation in the terrestrial biotic system. *The Prince Edward Islands: Land-Sea Interactions in a Changing World* (ed. by S.L. Chown and P.W. Froneman), pp. 245-276. African SunMedia, Stellenbosch.

 pie.neg

Pie Chart, considering negative values as zeros

Description

Plots a pie chart, considering negative values as zeros, for the purpose of illustrating variation partitioning.

Usage

```
pie.neg(
  x,
  labels = names(x),
  edges = 200,
  radius = 0.8,
  clockwise = FALSE,
  init.angle = if (clockwise) 90 else 0,
  density = NULL,
  angle = 45,
  col = NULL,
  border = NULL,
  lty = NULL,
  main = NULL,
  warning = TRUE,
  ...
)
```

Arguments

x	A vector of non-negative numerical quantities. The values in x are displayed as the areas of pie slices.
labels	One or more expressions or character strings giving names for the slices. Other objects are coerced by as.graphicsAnnot . For empty or NA (after coercion to character) labels, no label nor pointing line is drawn.
edges	The circular outline of the pie is approximated by a polygon with this many edges.
radius	The pie is drawn centered in a square box whose sides range from -1 to 1. If the character strings labeling the slices are long it may be necessary to use a smaller radius.
clockwise	Logical indicating if slices are drawn clockwise or counter clockwise (i.e., mathematically positive direction, used by default).
init.angle	number specifying the starting angle (in degrees) for the slices. Defaults to 0 (i.e., '3 o'clock') unless clockwise is true where init.angle defaults to 90 (degrees), (i.e., '12 o'clock').

density	The density of shading lines, in lines per inch. The default value of NULL means that no shading lines are drawn. Non-positive values of density also inhibit the drawing of shading lines.
angle	The slope of shading lines, given as an angle in degrees (counter-clockwise).
col	A vector of colors to be used in filling or shading the slices. If missing a set of 6 pastel colours is used, unless density is specified when par("fg") is used.
border, lty	(possibly vectors) arguments passed to polygon which draws each slice.
main	An overall title for the plot.
warning	Boolean value. Set to FALSE to avoid displaying a warning if some values are negative and set to 0.
...	Graphical parameters can be given as arguments to pie. They will affect the main title and labels only.

Details

This function is identical to the function `pie` in `{graphics}`, except that it considers all negative values as zeros, to allow for plotting variation partitioning outputs. The original `pie` function returns an error when negative values are present. However, variation partitioning can return negative values, which can then be treated as zeros (Legendre & Legendre, 2008). This function allows direct use of the results from `Zeta.varpart` without editing the data.

References

- Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988). *The new S language*. Wadsworth & Brooks/Cole.
- Cleveland, W. S. (1985). *The elements of graphing data*. Wadsworth: Monterey, CA, USA.
- Legendre, P. & Legendre, L.F. (2012). *Numerical ecology*, 3rd English edition. Elsevier Science BV, Amsterdam.

See Also

`pie`, `Zeta.varpart`

Examples

```
pie.neg(rep(1, 24), col = rainbow(24), radius = 0.9)
```

Plot.ispline

Plots I-splines for Multi-Site Generalised Dissimilarity Modelling

Description

Plots I-splines computed by `Return.ispline`, or calls `Return.ispline` if the outputs from `Zeta.msgdm` are provided before plotting.

Usage

```
Plot.ispline(
  isplines = NULL,
  msgdm,
  data.env,
  distance = FALSE,
  biotic = 0,
  pch = NULL,
  lty = NULL,
  legend = TRUE,
  lwd = 1,
  cex = 1,
  num.quantiles = 11
)
```

Arguments

<code>isplines</code>	Output of function <code>Return.ispline</code> .
<code>msgdm</code>	Output of function <code>Zeta.msgdm</code> computed with <code>reg.type = ispline</code> .
<code>data.env</code>	Site-by-variable data frame used for the computation of <code>msgdm</code> , with sites as rows and environmental variables as columns.
<code>distance</code>	Boolean, indicates is distance was used in the computation of <code>msgdm</code> .
<code>biotic</code>	Boolean, indicates is zeta diversity from another community was used in the computation of <code>msgdm</code> .
<code>pch</code>	Shapes of the points to be used in the plotting. If nothing is provided, <code>pch</code> is a sequence of integers from 1 to the number of variables used for the computation of <code>msgdm</code> .
<code>lty</code>	Line types to be used in the plotting. If nothing is provided, <code>lty</code> is a sequence of integers from 1 to the number of variables used for the computation of <code>msgdm</code> .
<code>legend</code>	Boolean, indicates if the legend must be drawn.
<code>lwd</code>	Line width.
<code>cex</code>	Point size.
<code>num.quantiles</code>	Number of points to plot on the I-splines. Default is 11 to plot a point every 10 percents of the range of values.

Value

`Plot.ispline` returns a data frame with the same number of rows as `dat` and `ncol(dat) * (order.ispline + kn.ispline)` columns.

References

- Ramsay, J. O. (1988). Monotone regression splines in action. *Statistical Science*, 425-441.
- Ferrier, S., Manion, G., Elith, J., & Richardson, K. (2007). Using generalized dissimilarity modelling to analyse and predict patterns of beta diversity in regional biodiversity assessment. *Diversity and Distributions*, 13(3), 252-264.

See Also[Zeta.msgdm](#)**Examples**

```
utils::data(Marion.species)
xy.marion <- Marion.species[1:2]
data.spec.marion <- Marion.species[3:33]

utils::data(Marion.env)
data.env.marion <- Marion.env[3]

zeta.ispline <- Zeta.msgdm(data.spec.marion, data.env.marion, xy.marion, sam = 100,
  order = 3, normalize = "Jaccard", reg.type = "ispline")
zeta.ispline
zeta.ispline.r <- Return.ispline(zeta.ispline, data.env.marion, distance = TRUE)
zeta.ispline.r

dev.new()
Plot.ispline(isplines = zeta.ispline.r, distance = TRUE)

dev.new()
Plot.ispline(msgdm = zeta.ispline, data.env = data.env.marion, distance = TRUE)
```

Plot.zeta.ddecay *Zeta distance-decay plotting*

Description

Plots the output of the function `Zeta.ddecay`.

Usage

```
Plot.zeta.ddecay(zeta.ddecay)
```

Arguments

`zeta.ddecay` A list produced by the function `Zeta.ddecay`.

Value

A plot of the zeta distance-decay with distance on the x-axis and the value of zeta on the y-axis.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#), [Zeta.ddecay](#),
[Zeta.ddecays](#)

Examples

```
utils::data(bird.spec.coarse)
xy.bird <- bird.spec.coarse[,1:2]
data.spec.bird <- bird.spec.coarse[,3:193]

dev.new()
zeta.ddecay.bird <- Zeta.ddecay(xy.bird, data.spec.bird, sam = 100, order = 3,
  confint.level = 0.95, plot=FALSE)
Plot.zeta.ddecay(zeta.ddecay.bird)

#####

utils::data(Marion.species)
xy.marion <- Marion.species[,1:2]
data.spec.marion <- Marion.species[,3:33]

zeta.ddecay.marion <- Zeta.ddecay(xy.marion, data.spec.marion, sam = 100, order = 3,
  confint.level = 0.95, trsf = "log", normalize = "Jaccard", plot=FALSE)
dev.new()
Plot.zeta.ddecay(zeta.ddecay.marion)
```

Plot.zeta.ddecays *Zeta distance-decay plotting for multiple orders*

Description

Plots the output of the function `Zeta.ddecays`.

Usage

```
Plot.zeta.ddecays(zeta.ddecays)
```

Arguments

`zeta.ddecays` A list produced by the function `Zeta.ddecays`.

Value

A plot of the zeta distance-decay with the orders on the x-axis and the slope of the linear distance-decays on the y-axis.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#), [Zeta.ddecays](#), [Zeta.ddecay](#), [Plot.zeta.ddecay](#)

Examples

```
utils::data(bird.spec.coarse)
xy.bird <- bird.spec.coarse[,1:2]
data.spec.bird <- bird.spec.coarse[,3:193]

dev.new()
zeta.ddecays.bird <- Zeta.ddecays(xy.bird, data.spec.bird, sam = 100, orders = 2:5,
  plot = FALSE, confint.level = 0.95)
Plot.zeta.ddecays(zeta.ddecays.bird)

#####

utils::data(Marion.species)
xy.marion <- Marion.species[,1:2]
data.spec.marion <- Marion.species[,3:33]

dev.new()
zeta.ddecays.marion <- Zeta.ddecays(xy.marion, data.spec.marion, sam = 100,
  orders = 2:5, plot = FALSE, confint.level = 0.95)
Plot.zeta.ddecays(zeta.ddecays.marion)
```

Plot.zeta.decline *Zeta diversity decline plotting*

Description

Plots the output of the functions `Zeta.decline.mc` and `Zeta.decline.ex`.

Usage

```
Plot.zeta.decline(zeta, sd.plot = TRUE, arrange.plots = TRUE)
```

Arguments

<code>zeta</code>	A list produced by the function <code>Zeta.decline.mc</code> or <code>Zeta.decline.ex</code> .
<code>sd.plot</code>	Boolean value (TRUE or FALSE) indicating if the standard deviation of each zeta diversity value must be plotted.
<code>arrange.plots</code>	Boolean value (TRUE or FALSE) indicating if the graphics device must be divided into 4 subplots.

Value

A plot of the zeta decline with 4 subplots displaying (i) the raw decline, (ii) the ratios of the zeta values (computed as ζ_i/ζ_{i-1}), (iii) the fit in a log plot and (iv) the fit in a log-log plot.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#)

Examples

```
utils::data(bird.spec.coarse)
xy.bird <- bird.spec.coarse[1:2]
data.spec.bird <- bird.spec.coarse[3:193]

dev.new(width = 12, height = 4)
zeta.bird <- Zeta.decline.mc(data.spec.bird, orders = 1:5, sam=100, plot = FALSE)
Plot.zeta.decline(zeta.bird)

#####

utils::data(Marion.species)
xy.marion <- Marion.species[1:2]
data.spec.marion <- Marion.species[3:33]

dev.new(width = 12, height = 4)
zeta.marion <- Zeta.decline.ex(data.spec.marion, orders = 1:5, plot = FALSE)
Plot.zeta.decline(zeta.marion)
```

Plot.zeta.scale.min.dist

Plotting of zeta diversity scaling with sample grain dependency based on the minimum distance between sites

Description

Plots the output of the function `Zeta.scale.min.dist`.

Usage

```
Plot.zeta.scale.min.dist(
  zeta.scale.irreg,
  size.init = 1,
  add = FALSE,
  ylim = NULL,
  col = "black"
)
```

Arguments

zeta.scale.irreg	A list generated by the function <code>Zeta.scale.min.dist</code> .
size.init	Initial size of the plots before aggregation.
add	Boolean value indicating if the graph must be plotted in a new graphics device or added to the active one.
ylim	Numeric vectors of length 2, giving the range of y values.
col	String indicating the color of the graph.

Value

A plot of the zeta diversity scaling with the mapping grain m (the number of sites combined to generate data at a coarser grain) on the x-axis and the value of zeta on the y-axis.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

Scheiner S.M., Chiarucci A., Fox G.A., Helmus M.R., McGlenn D.J. & Willig M.R. (2011). The underpinnings of the relationship of species richness with space and time. *Ecological Monographs*, 81, 195-213.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#),
[Zeta.scale.min.dist](#), [rescale.regular](#), [Zeta.scale.regular](#), [rescale.regular](#),
[Plot.zeta.scale.regular](#)

Examples

```
utils::data(Marion.species)
xy.marion <- Marion.species[,1:2]
data.spec.marion <- Marion.species[,3:33]

zeta.scale.irreg.species <- Zeta.scale.min.dist(xy.marion, data.spec.marion, m = 1:3,
  order = 3, reorder = 3, sam = 50, normalize = "Jaccard", plot=FALSE)
dev.new()
Plot.zeta.scale.min.dist(zeta.scale.irreg.species)
```

Plot.zeta.scale.regular

Plotting of zeta diversity scaling with sample grain using hierarchical increases in grain size

Description

Plots the output of the function `Zeta.scale.regular`.

Usage

```
Plot.zeta.scale.regular(
  zeta.scale.reg,
  size.init = 1,
  add = FALSE,
  ylim = NULL,
  col = "black"
)
```

Arguments

<code>zeta.scale.reg</code>	A list generated by the function <code>Zeta.scale.regular</code> .
<code>size.init</code>	initial Size of the plots before aggregation.
<code>add</code>	Boolean value indicating if the graph must be plotted in a new graphics device or added to the active one.
<code>ylim</code>	Numeric vectors of length 2, giving the range of y values.
<code>col</code>	String indicating the color of the graph.

Value

A plot of the zeta diversity scaling with the mapping grain $n \times n$ (the number of sites combined to generate data at a coarser grain) on the x-axis and the value of zeta on the y-axis.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

Scheiner S.M., Chiarucci A., Fox G.A., Helmus M.R., McGlenn D.J. & Willig M.R. (2011). The underpinnings of the relationship of species richness with space and time. *Ecological Monographs*, 81, 195-213.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#),
[Zeta.scale.regular](#), [Zeta.scale.min.dist](#), [rescale.regular](#),
[Plot.zeta.scale.min.dist](#)

Examples

```

utils::data(bird.spec.fine)
xy.bird <- bird.spec.fine[1:400,1:2]
data.spec.bird <- bird.spec.fine[1:400,3:192]

##sam = 25 is used here for fast execution, but a higher value is advised
zeta.scale.reg <- Zeta.scale.regular(xy.bird, data.spec.bird, n = 1:3, order = 3,
  sam = 25, normalize = "Jaccard",plot=FALSE)
dev.new()
Plot.zeta.scale.regular(zeta.scale.reg)

```

 Predict.msgdm

Predict zeta values for new environmental and distance data

Description

Predict the zeta values for new environmental and distance data from the models returned by Zeta.msgdm.

Usage

```
Predict.msgdm(model.msgdm, reg.type, newdata, type = "response")
```

Arguments

model.msgdm	A model returned by Zeta.msgdm. The class of the model depends on the type of regression used in Zeta.msgdm.
reg.type	Type of regression used in Zeta.msgdm. Options are "glm" for generalised linear models, "ngls" for negative linear models, "gam" for generalised additive models, "scam" for shape constrained additive models (with monotonic decreasing by default), and "ispline" for I-spline models (forcing monotonic decreasing), as recommended in generalised dissimilarity modelling by Ferrier <i>et al.</i> (2007).
newdata	A data frame with the new environmental and distance data. The names of the columns must be the same as the names used in the data frame used in Zeta.msgdm. For I-splines, the data frame must be generated beforehand from the original data by the function Ispline .
type	The type of prediction required, as for predict.glm. The default is on the scale of the response variable; the alternative "link" is on the scale of the linear predictors.

Value

Predict.msgdm returns a vector of predicted zeta values.

References

- Ramsay, J. O. (1988). Monotone regression splines in action. *Statistical Science*, 425-441.
- Ferrier, S., Manion, G., Elith, J., & Richardson, K. (2007). Using generalized dissimilarity modelling to analyse and predict patterns of beta diversity in regional biodiversity assessment. *Diversity and Distributions*, 13(3), 252-264.

See Also

[Zeta.msgdm](#)

Examples

```
utils::data(bird.spec.fine)
xy.bird <- bird.spec.fine[1:500,1:2]
data.spec.bird <- bird.spec.fine[1:500,3:192]
utils::data(bird.env.fine)
data.env.bird <- bird.env.fine[1:500,3:9]

zeta.glm <- Zeta.msgdm(data.spec.bird, data.env.bird, sam = 100, order = 3)
newdata <- data.frame(matrix(NA,100,ncol(data.env.bird)))
names(newdata) <- names(data.env.bird)
for(z in 1:100){
  samp <- sample(1:104, 3, replace = FALSE)
  newdata[z,] <- apply(apply(bird.env.fine[501:604,3:9][samp,], 2,
    stats::dist), 2, mean)
}
##rescale the data like during MS-GDM
newdata <- newdata/matrix(rep(zeta.glm$rescale.factor,100),
  100,length(zeta.glm$rescale.factor),byrow=TRUE)
new.zeta.glm <- Predict.msgdm(model.msgdm = zeta.glm$model, reg.type = "glm",
  newdata = newdata)

zeta.ngls <- Zeta.msgdm(data.spec.bird, data.env.bird, sam = 100, order = 3,
  reg.type = "ngls", normalize = "Jaccard")
newdata <- data.frame(matrix(NA,100,ncol(data.env.bird)))
names(newdata) <- names(data.env.bird)
for(z in 1:100){
  samp <- sample(1:104, 3, replace = FALSE)
  newdata[z,] <- apply(apply(bird.env.fine[501:604,3:9][samp,], 2, stats::dist),
    2, mean)
}
##rescale the data like during MS-GDM
newdata <- newdata/matrix(rep(zeta.ngls$rescale.factor,100),
  100,length(zeta.ngls$rescale.factor),byrow=TRUE)
new.zeta.ngls <- Predict.msgdm(model.msgdm = zeta.ngls$model, reg.type = "ngls",
  newdata = newdata)
```

Reg.ispline	<i>Perform an I-spline regression</i>
-------------	---------------------------------------

Description

Evaluates the I-splines for all variables of a data frame of predictor variables, and perform a generalised linear regression with constraint on the parameters.

Usage

```
Reg.ispline(
  response,
  predictor,
  order.ispline = 2,
  kn.ispline = 1,
  family = stats::gaussian(),
  method.glm = "glm.fit.cons",
  cons = 1,
  cons.inter = 1,
  control = list(),
  Plot = TRUE,
  lty = NULL,
  lwd = 1
)
```

Arguments

response	A vector of numeric values representing the response variable.
predictor	A data frame of numeric variables representing the predictors.
order.ispline	Order of the I-spline.
kn.ispline	Number of knots in the I-spline.
family	A description of the error distribution and link function to be used in the glm model (see family for details of family functions).
method.glm	Method used in fitting the generalised linear model. The default method "glm.fit.cons" is an adaptation of method <code>glm.fit2</code> from package <code>glm2</code> using a constrained least squares regression in the reweighted least squares. Another option is "glm.fit2", which calls <code>glm.fit2</code> ; see help documentation for <code>glm.fit2</code> in package <code>glm2</code> .
cons	type of constraint in the glm if <code>method.glm = "glm.fit.cons"</code> . Default is 1 for positive coefficients on the predictors. The other option is -1 for negative coefficients on the predictors.
cons.inter	type of constraint for the intercept. Default is 1 for positive intercept, suitable for Gaussian family. The other option is -1 for negative intercept, suitable for binomial family.
control	As for glm .

Plot	Boolean value indicating if the I-splines must be plotted.
lty	Line types to be used in the plotting. If nothing is provided, lty is a sequence of integers from 1 to the number of variables used for the computation of msgdm.
lwd	Line width.

Details

Reg.ispline performs a non-linear regression using a combination of GLM and I-splines. It can, for example, be used to compare regression outputs when using MS-GDM with I-splines on environmental variables and biotic variables as in `Zetya.msgdm` to the same regression approach without environmental variables.

Value

Reg.ispline returns a list of the following elements:

splines	A data frame in which each columns contains the value resulting from the transformation of the predictors into individual I-splines. The number of columns of splines is the number of predictors times the number of splines (determined as the sum of <code>order.ispline</code> and <code>kn.ispline</code>).
spline	A data frame in which each columns contains the value resulting from the combinations of the individual I-splines. This combination is obtained by multiplying the coefficients of <code>model</code> and the values of the individual I-splines <code>splines</code>
.	.
model	A <code>glm</code> model using <code>response</code> as the response variable, and <code>splines</code> as the predictors
.	.

References

Ramsay, J. O. (1988). Monotone regression splines in action. *Statistical Science*, 425-441.

See Also

[Zeta.msgdm,I spline](#)

Examples

```
utils::data(Marion.species)
xy.marion <- Marion.species[1:2]

data.spec.marion <- Marion.species[3:33]

##random other communities
data.spec.marion2a <- data.spec.marion
data.spec.marion2a[which(data.spec.marion2a==1,arr.ind=TRUE)] <- 0
for(i in 1:ncol(data.spec.marion2a))
```



```

data.spec.marion2a[sample(nrow(data.spec.marion2a),8),i] <- 1
data.spec.marion2b <- data.spec.marion
data.spec.marion2b[which(data.spec.marion2b==1,arr.ind=TRUE)] <- 0
for(i in 1:ncol(data.spec.marion2b))
data.spec.marion2b[sample(nrow(data.spec.marion2b),8),i] <- 1

dat.spec.tot <- list(data.spec.marion,data.spec.marion2a,data.spec.marion2b)
zeta.tot <- Zeta.order.mc.mult(data.spec=dat.spec.tot,order=3,sam=200)
zeta.splines <- Ispline(zeta.tot$zeta.val[,2:3])
data.tot <- data.frame(zeta.val=zeta.tot$zeta.val[,1],zeta.splines$splines)

dev.new()
Reg.ispline(response = zeta.tot$zeta.val[,1], predictor = zeta.tot$zeta.val[,2:3], lwd=2, cons=1)

```

rescale.min.dist

*Rescaling of data based on the minimum distance between sites***Description**

Combines sites based on the minimum distance between them.

Usage

```

rescale.min.dist(
  xy,
  data.spec,
  data.env = NULL,
  m,
  distance.type = "Euclidean",
  dist.custom = NULL,
  method = "mean",
  shuffle = FALSE
)

```

Arguments

xy	Site-by-coordinate data frame, with sites as rows and coordinates as columns.
data.spec	Site-by-species presence-absence data frame, with sites as rows and species as columns.
data.env	Site-by-variable data frame, with sites as rows and environmental variables as columns.
m	Mapping grain (the number of sites combined to generate data at a coarser grain). The m closest sites are grouped together.
distance.type	Method to compute distance. Default is "Euclidean", for Euclidean distance. The other options are (i) "ortho" for orthodromic distance, if xy correspond to longitudes and latitudes (orthodromic distance is computed using the geodist function from package geodist); and (ii) "custom", in which case the user must provide a distance matrix for dist.custom.

dist.custom	Distance matrix provided by the user when distance.type = "custom".
method	Name of a function (as a string) indicating how to combine the coordinates and the environmental variables. It can be a basic R-function such as "mean" or "max", but also a custom function.
shuffle	Boolean value (TRUE or FALSE) indicating if the order of the sites must be randomised, which can have an impact on the outputs if some distances are equal.

Details

The nearest neighbouring sites (plots, quadrates, or areas of varying shapes) are grouped as spatial clusters of 2, 3, 4, etc. sites, based on the minimum distance between them. Since the procedure is based on the relative distance between sites, the site order can have an impact on the output. This function is suitable for both regularly and irregularly spaced sites, contiguous or non contiguous. For regularly spaced sites, the use of [rescale.regular](#) is recommended.

Value

rescale.min.dist returns a data frame with the rescaled data.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

Scheiner S.M., Chiarucci A., Fox G.A., Helmus M.R., McGlenn D.J. & Willig M.R. (2011). The underpinnings of the relationship of species richness with space and time. *Ecological Monographs*, 81, 195-213.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#),
[Zeta.scale.min.dist](#), [Zeta.scale.regular](#), [rescale.regular](#)

Examples

```
utils::data(Marion.species)
xy.marion <- Marion.species[,1:2]
data.spec.marion <- Marion.species[,3:33]

data.rescale <- rescale.min.dist(xy.marion, data.spec.marion, m=2)
```

rescale.regular	<i>Rescaling of data following a hierarchical increase in grain size</i>
-----------------	--

Description

Increases grain by hierarchically nesting regularly spaced sites.

Usage

```
rescale.regular(xy, data.spec, data.env = NULL, method = "mean", n)
```

Arguments

xy	Site-by-coordinate data frame, with sites as rows and coordinates as columns.
data.spec	Site-by-species presence-absence data frame, with sites as rows and species as columns.
data.env	Site-by-variable data frame, with sites as rows and environmental variables as columns.
method	Name of a function (as a string) indicating how to combine the coordinates and the environmental variables. It can be a basic R-function such as "mean" or "max", but also a custom function.
n	Mapping grain (the number of sites combined to generate data at a coarser grain). Regularly spaced sites are grouped as $n \times n$ sites.

Details

The sites (plots or quadrates) are aggregated as nearest neighbouring groups of $n \times n$ sites, using a nested approach, starting from the lowest x and y, to increase the grain. The sites can be spatially contiguous or discontinuous, as long as they are regularly spaced. This function is not suitable for irregularly spaced sites. If the total number of sites is not a multiple of $n \times n$, the extra sites are discarded.

Value

rescale.regular returns a data frame with the rescaled data.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

Scheiner S.M., Chiarucci A., Fox G.A., Helmus M.R., McGlenn D.J. & Willig M.R. (2011). The underpinnings of the relationship of species richness with space and time. *Ecological Monographs*, 81, 195-213.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#), [Zeta.scale.regular](#), [Zeta.scale.min.dist](#), [rescale.min.dist](#)

Examples

```
utils::data(bird.spec.fine)
xy.bird <- bird.spec.fine[1:2]
data.spec.bird <- bird.spec.fine[3:192]

data.rescale <- rescale.regular(xy.bird, data.spec.bird, n = 4)
```

Return.ispline	<i>Computing splines coordinates from I-spline-based multi-site generalised dissimilarity modelling</i>
----------------	---

Description

Stores the coordinates of the I-splines resulting from `Zeta.msgdm` for plotting.

Usage

```
Return.ispline(msgdm, data.env, distance = FALSE, biotic = 0)
```

Arguments

<code>msgdm</code>	Output of function <code>Zeta.msgdm</code> computed with <code>reg.type = ispline</code> .
<code>data.env</code>	Site-by-variable data frame used for the computation of <code>msgdm</code> , with sites as rows and environmental variables as columns.
<code>distance</code>	Boolean, indicates if distance was used in the computation of <code>msgdm</code> .
<code>biotic</code>	Integer, indicates the number of other groups of taxa for which zeta diversity was computed and used in the computation of <code>msgdm</code> .

Details

`Return.ispline` allows to store the same number of coordinates for all I-splines, to average replicates and obtain confidence intervals.

Value

`Return.ispline` returns a list containing the following components used to plot the I-splines:

<code>env</code>	A data frame containing the rescaled environmental (numeric and factor), distance and biotic x-values.
<code>I-spline</code>	A data frame containing the I-spline values corresponding to the rescaled environmental (numeric and factor), distance and biotic x-values.

See Also

[Zeta.msgdm, Ispline](#)

Examples

```
utils::data(Marion.species)
xy.marion <- Marion.species[1:2]
data.spec.marion <- Marion.species[3:33]

utils::data(Marion.env)
data.env.marion <- Marion.env[3]

zeta.ispline <- Zeta.msgdm(data.spec.marion, data.env.marion, xy.marion, sam = 100,
  order = 3, normalize = "Jaccard", reg.type = "ispline")
zeta.ispline
zeta.ispline.r <- Return.ispline(zeta.ispline, data.env.marion, distance = TRUE)
zeta.ispline.r

dev.new()
Plot.ispline(isplines = zeta.ispline.r, distance = TRUE)

dev.new()
Plot.ispline(msgdm = zeta.ispline, data.env = data.env.marion, distance = TRUE)
```

Zeta.ddecay

Zeta distance decay for a specific number of assemblages or sites

Description

Computes the distance decay of zeta diversity for a specific order (number of assemblages or sites), using either a generalised linear model with possible constraint on the coefficients, a generalised additive model, or a shape constrained additive model.

Usage

```
Zeta.ddecay(
  xy,
  data.spec,
  order = 2,
  sam = 1000,
  distance.type = "Euclidean",
  dist.custom = NULL,
  method = "mean",
  reg.type = "glm",
  family = stats::gaussian(),
  method.glm = "glm.fit.cons",
```

```

cons = -1,
cons.inter = 1,
confint.level = 0.95,
kn = -1,
bs = "mpd",
trsf = "NULL",
cutoff = NULL,
rescale = FALSE,
normalize = FALSE,
empty.row = "remove",
plot = TRUE
)

```

Arguments

<code>xy</code>	Site-by-coordinate data frame, with sites as rows and coordinates as columns.
<code>data.spec</code>	Site-by-species presence-absence data frame, with sites as rows and species as columns.
<code>order</code>	Specific number of assemblages or sites at which zeta diversity is computed.
<code>sam</code>	Number of samples for which the zeta diversity is computed.
<code>distance.type</code>	Method to compute distance. Default is "Euclidean", for Euclidean distance. The other options are (i) "ortho" for orthodromic distance, if <code>xy</code> correspond to longitudes and latitudes (orthodromic distance is computed using the <code>geodist</code> function from package <code>geodist</code>); and (ii) "custom", in which case the user must provide a distance matrix for <code>dist.custom</code> .
<code>dist.custom</code>	Distance matrix provided by the user when <code>distance.type = "custom"</code> .
<code>method</code>	Name of a function (as a string) indicating how to combine the pairwise differences and distances for more than 3 sites. It can be a basic R-function such as "mean" or "max", but also a custom function.
<code>reg.type</code>	Type of regression. Options are "glm" for generalised linear models "gam" for generalised additive models and "scam" for shape constrained additive models (with monotonic decreasing by default).
<code>family</code>	A description of the error distribution and link function to be used in the <code>glm</code> , <code>gam</code> and <code>scam</code> models (see family for details of family functions).
<code>method.glm</code>	Method used in fitting the generalised linear model. The default method "glm.fit.cons" is an adaptation of method <code>glm.fit2</code> from package <code>glm2</code> using a negative least squares regression in the reweighted least squares. Another option is "glm.fit2", which calls <code>glm.fit2</code> ; see help documentation for <code>glm.fit2</code> in package <code>glm</code> .
<code>cons</code>	type of constraint in the <code>glm</code> if <code>method.glm = "glm.fit.cons"</code> . Default is -1 for negative coefficients on the predictors. The other option is 1 for positive coefficients on the predictors.
<code>cons.inter</code>	type of constraint for the intercept. Default is 1 for positive intercept, suitable for Gaussian family. The other option is -1 for negative intercept, suitable for binomial family.

<code>confint.level</code>	Percentage for the confidence intervals of the coefficients from the generalised linear models.
<code>kn</code>	Number of knots in the GAM and SCAM. Default is -1 for determining <code>kn</code> automatically using Generalized Cross-validation.
<code>bs</code>	A two-letter character string indicating the (penalized) smoothing basis to use in the scam model. Default is "mpd" for monotonic decreasing splines. see smooth.terms for an overview of what is available.
<code>trsf</code>	Name of a function (as a string) indicating how to transform distance.
<code>cutoff</code>	If specified, maximum distance value for which the linear regression must be performed.
<code>rescale</code>	Boolean value (TRUE or FALSE) indicating if the zeta values should be divided by ζ_1 , to get a range of values between 0 and 1. Has no effect if <code>normalize != FALSE</code> .
<code>normalize</code>	Indicates if the zeta values for each sample should be divided by the total number of species for this specific sample (<code>normalize = "Jaccard"</code>), by the average number of species per site for this specific sample (<code>normalize = "Sorensen"</code>), or by the minimum number of species in the sites of this specific sample (<code>normalize = "Simpson"</code>). Default value is FALSE, indicating that no normalization is performed.
<code>empty.row</code>	Determines how to handle empty rows, i.e. sites with no species. Such sites can cause underestimations of zeta diversity, and computation errors for the normalized version of zeta due to divisions by 0. Options are "empty" to let the data untreated, "remove" to remove the empty rows, 0 to set the normalized zeta to 0 when zeta is divided by 0 during normalization (sites share no species, so are completely dissimilar), and 1 to set the normalized zeta to 1 when zeta is divided by 0 during normalization (i.e. sites are perfectly similar).
<code>plot</code>	Boolean value (TRUE or FALSE) indicating if the outputs must be plotted.

Value

`Zeta.ddecay` returns a list containing the following components:

<code>order</code>	The order of zeta for which the distance decay was computed.
<code>reg.type</code>	A character string indicating the type of regression that was performed.
<code>reg</code>	An object whose class depends on the type of regression (<code>glm</code> , <code>gam</code> or <code>scam</code>), corresponding to the regression over distance for the number of assemblages or sites specified in <code>'order'</code> .
<code>confint</code>	The confidence intervals for the coefficients from the generalised linear model. <code>confint</code> is not generated for generalised additive models and shape constrained additive models.
<code>zeta.val</code>	The values of zeta for the sampled sites used in the regression.
<code>distance</code>	The distances for the sampled sites used in the regression.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#), [Zeta.ddecays](#),
[Plot.zeta.ddecay](#)

Examples

```
utils::data(bird.spec.coarse)
xy.bird <- bird.spec.coarse[,1:2]
data.spec.bird <- bird.spec.coarse[,3:193]

dev.new()
zeta.ddecay.bird <- Zeta.ddecay(xy.bird, data.spec.bird, sam = 100, order = 3,
  method.glm = "glm.fit2", confint.level = 0.95)

dev.new()
zeta.ddecay.bird <- Zeta.ddecay(data.spec=data.spec.bird, distance.type = "custom",
  dist.custom = as.matrix(dist(xy.bird)), cutoff = 800000, sam = 100, order = 3,
  reg.type = "gam", confint.level = 0.95)

#####

utils::data(Marion.species)
xy.marion <- Marion.species[,1:2]
data.spec.marion <- Marion.species[,3:33]

dev.new()
zeta.ddecay.marion <- Zeta.ddecay(xy.marion, data.spec.marion, sam = 100, order = 3,
  method.glm = "glm.fit2", confint.level = 0.95, trsf = "log", normalize = "Jaccard")
```

Zeta.ddecays

Zeta distance decay for a range of numbers of assemblages or sites

Description

Computes the distance decay of zeta diversity for a range of orders (number of assemblages or sites), using generalised linear models.

Usage

```
Zeta.ddecays(
  xy,
  data.spec,
  orders = 2:10,
  sam = 1000,
  family = stats::gaussian(),
  distance.type = "Euclidean",
  dist.custom = NULL,
```



```

method = "mean",
confint.level = 0.95,
trsf = "NULL",
cutoff = NULL,
rescale = FALSE,
normalize = FALSE,
plot = TRUE
)

```

Arguments

<code>xy</code>	Site-by-coordinate data frame, with sites as rows and coordinates as columns.
<code>data.spec</code>	Site-by-species presence-absence data frame, with sites as rows and species as columns.
<code>orders</code>	Range of number of assemblages or sites at which zeta diversity is computed. All the orders must be strictly greater than 1.
<code>sam</code>	Number of samples for which the zeta diversity is computed.
<code>family</code>	A description of the error distribution and link function to be used in the generalised linear models (see family for details of family functions).
<code>distance.type</code>	Method to compute distance. Default is "Euclidean", for Euclidean distance. The other options are (i) "ortho" for orthodromic distance, if <code>xy</code> correspond to longitudes and latitudes (orthodromic distance is computed using the <code>geodist</code> function from package <code>geodist</code>); and (ii) "custom", in which case the user must provide a distance matrix for <code>dist.custom</code> .
<code>dist.custom</code>	Distance matrix provided by the user when <code>distance.type = "custom"</code> .
<code>method</code>	Name of a function (as a string) indicating how to combine the pairwise differences and distances for more than 3 sites. It can be a basic R-function such as "mean" or "max", but also a custom function.
<code>confint.level</code>	Percentage for the confidence intervals of the coefficients from the linear regression.
<code>trsf</code>	Name of a function (as a string) indicating how to transform distance. Default is "NULL" for the identity transformation.
<code>cutoff</code>	If specified, maximum distance value for which the linear regression must be performed.
<code>rescale</code>	Boolean value (TRUE or FALSE) indicating if the zeta values should be divided by ζ_1 , to get a range of values between 0 and 1. Has no effect if <code>normalize != FALSE</code> .
<code>normalize</code>	Indicates if the zeta values for each sample should be divided by the total number of species for this specific sample (<code>normalize = "Jaccard"</code>), by the average number of species per site for this specific sample (<code>normalize = "Sorensen"</code>), or by the minimum number of species in the sites of this specific sample (<code>normalize = "Simpson"</code>). Default value is FALSE, indicating that no normalization is performed.
<code>plot</code>	Boolean value (TRUE or FALSE) indicating if the outputs must be plotted.

Value

Zeta.ddecays returns a list containing the following components:

orders	Range of number of assemblages or sites at which zeta diversity was computed.
coefs	A vector of the coefficients from the generalised linear models for the numbers of sites specified by orders.
confint	The confidence intervals for the coefficients from the generalised linear models.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#), [Zeta.ddecay](#)

Examples

```
utils::data(bird.spec.coarse)
xy.bird <- bird.spec.coarse[,1:2]
data.spec.bird <- bird.spec.coarse[,3:193]

dev.new()
zeta.ddecays.bird <- Zeta.ddecays(xy.bird, data.spec.bird, sam = 100, orders = 2:5,
  plot = TRUE, confint.level = 0.95)

#####

utils::data(Marion.species)
xy.marion <- Marion.species[,1:2]
data.spec.marion <- Marion.species[,3:33]

dev.new()
zeta.ddecays.marion <- Zeta.ddecays(xy.marion, data.spec.marion, sam = 100,
  orders = 2:5, plot = TRUE, confint.level = 0.95)
```

Zeta.decline.ex

Expectation of zeta diversity decline

Description

Computes the expectation of zeta diversity, the number of species shared by multiple assemblages for a range of orders (number of assemblages or sites), using a formula based on the occupancy of the species, and fits the decline to an exponential and a power law relationship.

Usage

```
Zeta.decline.ex(
  data.spec,
  orders = 1:10,
  sd.correct = TRUE,
  confint.level = 0.95,
  sd.plot = TRUE,
  rescale = FALSE,
  empty.row = "empty",
  plot = TRUE
)
```

Arguments

<code>data.spec</code>	Site-by-species presence-absence data frame, with sites as rows and species as columns.
<code>orders</code>	Range of number of assemblages or sites for which zeta diversity is computed.
<code>sd.correct</code>	Boolean value (TRUE or FALSE) indicating if the standard deviation must be computed with an unbiased estimator (using the number of site combinations - 1 as the denominator) or not (using the number of site combinations as the denominator).
<code>confint.level</code>	Percentage for the confidence intervals of the coefficients from the regressions.
<code>sd.plot</code>	Boolean value (TRUE or FALSE) indicating if the standard deviation of each zeta diversity value must be plotted.
<code>rescale</code>	Boolean value (TRUE or FALSE) indicating if the zeta values should be divided by ζ_1 , to get a range of values between 0 and 1.
<code>empty.row</code>	Determines how to handle empty rows, i.e. sites with no species. Such sites can cause underestimations of zeta diversity. Options are "empty" to let the data untreated or "remove" to remove the empty rows.
<code>plot</code>	Boolean value (TRUE or FALSE) indicating if the outputs must be plotted.

Details

`Zeta.decline.ex` is much faster than `Zeta.decline.mc` to compute the exact value of zeta diversity when the number of species is lower than C_i^N , where N is the total number of sites and i is the order of zeta.

`sd.correct` should be set to TRUE if the assemblages represent a subsample of the whole system. It can be set to FALSE if the sampling is exhaustive, for example in case of a continuous regular grid covering the whole study area.

The exponential and the power law fit are performed using linear regressions on log-transformed data (only the zeta values are log-transformed for the exponential fit, and both the orders and the zeta values are log-transformed for the power law fit).

Value

Zeta.decline.ex returns a list containing the following components:

zeta.order	The number of assemblages or sites for which the zeta diversity was computed.
combinations	The number of possible combinations of sites for the chosen orders.
zeta.val	The zeta diversity values.
zeta.val.sd	The zeta diversity standard deviation values.
zeta.ratio	The ratio of zeta diversity values by the zeta diversity values at the lower order ζ_i/ζ_{i-1} .
zeta.exp	Object of class "lm", containing the output of the exponential regression.
zeta.exp.confint	The confidence intervals of the coefficients of the exponential regression.
zeta.pl	Object of class "lm", containing the output of the power law regression.
zeta.pl.confint	The confidence intervals of the coefficients of the power law regression.
aic	AIC values for zeta.exp and zeta.pl.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

McGeoch M. A., Latombe G., Andrew N. R., Nakagawa S., Nipperess D. A., Roige M., Marzinelli E. M., Campbell A. H., Verges A., Thomas T., Steinberg P. D., Selwood K. E., Henriksen M. V. & Hui C. (2019). Measuring continuous compositional change using decline and decay in zeta diversity. *Ecology*, 100(11), e02832.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.order.ex](#), [Plot.zeta.decline](#)

Examples

```
utils::data(bird.spec.coarse)
data.spec.bird <- bird.spec.coarse[,3:193]

dev.new(width = 12, height = 4)
zeta.bird <- Zeta.decline.ex(data.spec.bird, orders = 1:5)
zeta.bird

#####

utils::data(Marion.species)
xy.marion <- Marion.species[,1:2]
data.spec.marion <- Marion.species[,3:33]

dev.new(width = 12, height = 4)
zeta.marion <- Zeta.decline.ex(data.spec.marion, orders = 1:5)
zeta.marion
```

Zeta.decline.mc *Zeta diversity decline using Monte Carlo sampling*

Description

Computes zeta diversity, the number of species shared by multiple assemblages, for a range of orders (number of assemblages or sites), using combinations of sampled sites, and fits the decline to an exponential and a power law relationship.

Usage

```
Zeta.decline.mc(
  data.spec,
  xy = NULL,
  orders = 1:10,
  sam = 1000,
  sd.correct = TRUE,
  sd.correct.adapt = FALSE,
  confint.level = 0.95,
  sd.plot = TRUE,
  rescale = FALSE,
  normalize = FALSE,
  NON = FALSE,
  FPO = NULL,
  DIR = FALSE,
  empty.row = "empty",
  plot = TRUE,
  silent = TRUE
)
```

Arguments

data.spec	Site-by-species presence-absence data frame, with sites as rows and species as columns.
xy	Site coordinates. This is only used if NON = TRUE or DIR = TRUE.
orders	Range of number of assemblages or sites for which zeta diversity is computed.
sam	Number of samples for which the zeta diversity is computed for each number of assemblages or sites.
sd.correct	Boolean value (TRUE or FALSE) indicating if the standard deviation must be computed with an unbiased estimator (using the number of site combinations - 1 as the denominator) or not (using the number of site combinations as the denominator).
sd.correct.adapt	Boolean value (TRUE or FALSE) indicating if the standard deviation must be computed with an unbiased estimator (using the number of site combinations - 1 as the denominator) if sam is higher than the number of possible combinations,

	or not (using the number of site combinations as the denominator) if <code>sam</code> is lower than the number of possible combinations. If <code>sd.correct.adapt = TRUE</code> , it takes precedence over <code>sd.correct</code> .
<code>confint.level</code>	Percentage for the confidence intervals of the coefficients from the regressions.
<code>sd.plot</code>	Boolean value (TRUE or FALSE) indicating if the standard deviation of each zeta diversity value must be plotted.
<code>rescale</code>	Boolean value (TRUE or FALSE) indicating if the zeta values should be divided by ζ_1 , to get a range of values between 0 and 1. Has no effect if <code>normalize != FALSE</code> .
<code>normalize</code>	Indicates if the zeta values for each sample should be divided by the total number of species for this specific sample (<code>normalize = "Jaccard"</code>), by the average number of species per site for this specific sample (<code>normalize = "Sorensen"</code>), or by the minimum number of species in the sites of this specific sample (<code>normalize = "Simpson"</code>). Default value is FALSE, indicating that no normalization is performed.
<code>NON</code>	Boolean value (TRUE or FALSE) indicating if the number of species in common should only be counted for the nearest neighbours.
<code>FPO</code>	A vector with the coordinates of the fixed point origin from which the zeta diversity will be computed (overrides <code>NON</code>). In that case, ζ_1 is the number of species in the closest site to the FPO, ζ_2 is the number of species shared by the 2 closest sites, etc.
<code>DIR</code>	Boolean value (TRUE or FALSE) indicating if zeta diversity must be computed using a directed nearest neighbour scheme in the direction away from the FPO, starting from any site.
<code>empty.row</code>	Determines how to handle empty rows, i.e. sites with no species. Such sites can cause underestimations of zeta diversity, and computation errors for the normalized version of zeta due to divisions by 0. Options are "empty" to let the data untreated, "remove" to remove the empty rows, 0 to set the normalized zeta to 0 when zeta is divided by 0 during normalization (sites share no species, so are completely dissimilar), and 1 to set the normalized zeta to 1 when zeta is divided by 0 during normalization (i.e. sites are perfectly similar).
<code>plot</code>	Boolean value (TRUE or FALSE) indicating if the outputs must be plotted.
<code>silent</code>	Boolean value (TRUE or FALSE) indicating if messages must be printed.

Details

If the number of combinations of sites is lower than the value of the parameter `sam`, all the combinations are used and an exact solution is computed. In that case, using the number of site combinations as the denominator may be appropriate to compute the standard deviation, if all sites were sampled and the zeta values. This can be adjusted with parameters `sd.correct` and `sd.correct.adapt`.

`Zeta.decline.mc` is faster than `Zeta.decline.ex` to compute the exact value of zeta diversity when the number of species is higher than C_i^N , where N is the total number of sites and i is the order of zeta.

The exponential and the power law fit are performed using linear regressions on log-transformed data (only the zeta values are log-transformed for the exponential fit, and both the orders and the zeta values are log-transformed for the power law fit).

Zeta.decline.mc enables accomodating richness heterogeneity by setting `normalize = "Jaccard"`, `normalize = "Sorensen"` or `normalize = "Simpson"`. This cannot be performed by [Zeta.decline.ex](#).

Value

Zeta.decline.mc returns a list containing the following components:

<code>zeta.order</code>	The number of assemblages or sites for which the zeta diversity was computed.
<code>combinations</code>	The number of possible combinations of sites for the chosen orders.
<code>zeta.val</code>	The zeta diversity values.
<code>zeta.val.sd</code>	The zeta diversity standard deviation values.
<code>zeta.ratio</code>	The ratio of zeta diversity values by the zeta diversity values at the lower order ζ_i/ζ_{i-1} .
<code>zeta.exp</code>	Object of class "lm", containing the output of the exponential regression.
<code>zeta.exp.confint</code>	The confidence intervals of the coefficients of the exponential regression.
<code>zeta.pl</code>	Object of class "lm", containing the output of the power law regression.
<code>zeta.pl.confint</code>	The confidence intervals of the coefficients of the power law regression.
<code>aic</code>	AIC values for <code>zeta.exp</code> and <code>zeta.pl</code> .

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

See Also

[Zeta.decline.ex](#), [Zeta.order.ex](#), [Zeta.order.mc](#), [Plot.zeta.decline](#)

Examples

```
utils::data(bird.spec.coarse)
xy.bird <- bird.spec.coarse[,1:2]
data.spec.bird <- bird.spec.coarse[,3:193]

dev.new(width = 12, height = 4)
zeta.bird <- Zeta.decline.mc(data.spec.bird, xy.bird, orders = 1:5, sam = 100,
  NON = TRUE)
zeta.bird

#####

utils::data(Marion.species)
xy.marion <- Marion.species[,1:2]
data.spec.marion <- Marion.species[,3:33]
```

```
dev.new(width = 12, height = 4)
zeta.marion <- Zeta.decline.mc(data.spec.marion, orders = 1:5, sam = 100,
  normalize = "Jaccard")
zeta.marion
```

Zeta.msgdm

Multi-site generalised dissimilarity modelling for a set of environmental variables and distances

Description

Computes a regression model of zeta diversity for a given order (number of assemblages or sites) against a set of environmental variables and distances between sites. The different regression models available are generalised linear models, generalised linear models with negative constraints, generalised additive models, shape constrained additive models, and I-splines.

Usage

```
Zeta.msgdm(
  data.spec,
  data.env,
  xy = NULL,
  data.spec.pred = NULL,
  order = 1,
  sam = 1000,
  reg.type = "glm",
  family = stats::gaussian(),
  method.glm = "glm.fit.cons",
  cons = -1,
  cons.inter = 1,
  confint.level = 0.95,
  bs = "mpd",
  kn = -1,
  order.ispline = 2,
  kn.ispline = 1,
  distance.type = "Euclidean",
  dist.custom = NULL,
  rescale = FALSE,
  rescale.pred = TRUE,
  method = "mean",
  normalize = FALSE,
  silent = FALSE,
  empty.row = 0,
  control = list(),
  glm.init = FALSE
)
```


Arguments

<code>data.spec</code>	Site-by-species presence-absence data frame, with sites as rows and species as columns.
<code>data.env</code>	Site-by-variable data frame, with sites as rows and environmental variables as columns.
<code>xy</code>	Site coordinates, to account for distances between sites.
<code>data.spec.pred</code>	Site-by-species presence-absence data frame or list of data frames, with sites as rows and species as columns, for which zeta diversity will be computed and used as a predictor of the zeta diversity of <code>data.spec</code> .
<code>order</code>	Specific number of assemblages or sites at which zeta diversity is computed.
<code>sam</code>	Number of samples for which the zeta diversity is computed.
<code>reg.type</code>	Type of regression used in the multi-site generalised dissimilarity modelling. Options are "glm" for generalised linear models, "ngls" for negative linear models, "gam" for generalised additive models, "scam" for shape constrained additive models (with monotonic decreasing by default), and "ispline" for I-spline models (forcing monotonic decline), as recommended in generalised dissimilarity modelling by Ferrier <i>et al.</i> (2007).
<code>family</code>	A description of the error distribution and link function to be used in the glm, gam and scam models (see family for details of family functions).
<code>method.glm</code>	Method used in fitting the generalised linear model. The default method "glm.fit.cons" is an adaptation of method <code>glm.fit2</code> from package <code>glm2</code> using a constrained least squares regression (default is negative coefficients) in the reweighted least squares. Another option is "glm.fit2", which calls <code>glm.fit2</code> ; see help documentation for <code>glm.fit2</code> in package <code>glm2</code> .
<code>cons</code>	type of constraint in the glm if <code>method.glm = "glm.fit.cons"</code> . Default is -1 for negative coefficients on the predictors. The other option is 1 for positive coefficients on the predictors.
<code>cons.inter</code>	type of constraint for the intercept. Default is 1 for positive intercept, suitable for Gaussian family. The other option is -1 for negative intercept, suitable for binomial family.
<code>confint.level</code>	Percentage for the confidence intervals of the coefficients from the generalised linear models.
<code>bs</code>	A two-letter character string indicating the (penalized) smoothing basis to use in the scam model. Default is "mpd" for monotonic decreasing splines. see smooth.terms for an overview of what is available.
<code>kn</code>	Number of knots in the GAM and SCAM. Default is -1 for determining kn automatically using Generalized Cross-validation.
<code>order.ispline</code>	Order of the I-spline.
<code>kn.ispline</code>	Number of knots in the I-spline.
<code>distance.type</code>	Method to compute distance. Default is "Euclidean", for Euclidean distance. The other options are (i) "ortho" for orthodromic distance, if <code>xy</code> correspond to longitudes and latitudes (orthodromic distance is computed using the <code>geodist</code> function from package <code>geodist</code>); and (ii) "custom", in which case the user must provide a distance matrix for <code>dist.custom</code> .

<code>dist.custom</code>	Distance matrix provided by the user when <code>distance.type = "custom"</code> .
<code>rescale</code>	Boolean value (TRUE or FALSE) indicating if the zeta values should be divided by the total number of species in the dataset, to get a range of values between 0 and 1. Has no effect if <code>normalize != FALSE</code> .
<code>rescale.pred</code>	Boolean value (TRUE or FALSE) indicating if the spatial distances and differences in environmental variables should be rescaled between 0 and 1.
<code>method</code>	Name of a function (as a string) indicating how to combine the pairwise differences and distances for more than 3 sites. It can be a basic R-function such as "mean" or "max", but also a custom function.
<code>normalize</code>	Indicates if the zeta values for each sample should be divided by the total number of species for this specific sample (<code>normalize = "Jaccard"</code>), by the average number of species per site for this specific sample (<code>normalize = "Sorensen"</code>), or by the minimum number of species in the sites of this specific sample (<code>normalize = "Simpson"</code>). Default value is FALSE, indicating that no normalization is performed.
<code>silent</code>	Boolean value (TRUE or FALSE) indicating if warnings must be printed.
<code>empty.row</code>	Determines how to handle empty rows, i.e. sites with no species. Such sites can cause underestimations of zeta diversity, and computation errors for the normalized version of zeta due to divisions by 0. Options are "empty" to let the data untreated, "remove" to remove the empty rows, 0 to set the normalized zeta to 0 when zeta is divided by 0 during normalization (sites share no species, so are completely dissimilar), and 1 to set the normalized zeta to 1 when zeta is divided by 0 during normalization (i.e. sites are perfectly similar).
<code>control</code>	As for <code>glm</code> .
<code>glm.init</code>	Boolean value, indicating if the initial parameters for fitting the <code>glm</code> with constraint on the coefficients signs for <code>reg.type == "ispline"</code> should be initialised based on the correlation coefficients between the zeta values and the environmental difference or distance. <code>glm.init = TRUE</code> helps preventing the error message: <code>error: cannot find valid starting values: please specify some.</code>

Details

The environmental variables can be numeric or factorial.

If `order = 1`, the variables are used as such in the regression, and factorial variables must be dummy for the output of the regression to be interpretable.

For numeric variables, if `order > 1` the pairwise difference between sites is computed and combined according to `method`. For factorial variables, the distance corresponds to the number of unique values over the number of assemblages of sites specified by `order`.

If `xy = NULL`, `Zeta.msgdm` only uses environmental variables in the regression. Otherwise, it also computes and uses euclidian distance (average or maximum distance between multiple sites, depending on the parameters `method`) as an explanatory variable.

If `rescale.pred = TRUE`, zeta is regressed against the differences of values of the environmental variables divided by the maximum difference for each variable, to be rescaled between 0 and 1. If `!is.null(xy)`, distances between sites are also divided by the maximum distance. If `order = 1`, the

variables are transformed by first subtracting their minimum value, and dividing by the difference of their maximum and minimum values.

If `reg.type = "ispline"`, the variables are rescaled between 0 and 1 prior to computing the I-splines by subtracting their minimum value, and dividing by the difference of their maximum and minimum values.

Value

`Zeta.msgdm` returns a list whose component vary depending on the regression technique. The list can contain the following components:

<code>val</code>	Vector of zeta values used in the MS-GDM.
<code>predictors</code>	Data frame of the predictors used in the MS-GDM.
<code>range.min</code>	Vector containing the minimum values of the numeric variables, used for rescaling the variables between 0 and 1 for I-splines (see Details).
<code>range.max</code>	Vector containing the maximum values of the numeric variables, used for rescaling the variables between 0 and 1 for I-splines (see Details).
<code>rescale.factor</code>	Factor by which the predictors were divided if <code>rescale.pred = TRUE</code> and <code>order>1</code> .
<code>order.ispline</code>	The value of the original parameter, to be used in <code>Plot.ispline</code> .
<code>kn.ispline</code>	The value of the original parameter, to be used in <code>Plot.ispline</code> .
<code>model</code>	An object whose class depends on the type of regression (<code>glm</code> , <code>nnnpls</code> , <code>gam</code> or <code>scam</code> ; I-splines return an object of class <code>glm</code>), corresponding to the regression over distance for the number of assemblages or sites specified in <code>order</code> .
<code>confint</code>	The confidence intervals for the coefficients from generalised linear models with no constraint. <code>confint</code> is not generated for the other types of regression.
<code>vif</code>	The variance inflation factors for all the variables for the generalised linear regression. <code>vif</code> is not generated for the other types of regression.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

Ferrier, S., Manion, G., Elith, J., & Richardson, K. (2007). Using generalized dissimilarity modelling to analyse and predict patterns of beta diversity in regional biodiversity assessment. *Diversity and Distributions*, 13(3), 252-264.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#), [Predict.msgdm](#), [Ispline](#)

Examples

```

utils::data(bird.spec.coarse)
xy.bird <- bird.spec.coarse[1:2]
data.spec.bird <- bird.spec.coarse[3:193]
utils::data(bird.env.coarse)
data.env.bird <- bird.env.coarse[,3:9]

zeta.glm <- Zeta.msgdm(data.spec.bird, data.env.bird, sam = 100, order = 3)
zeta.glm
dev.new()
graphics::plot(zeta.glm$model)

zeta.ngls <- Zeta.msgdm(data.spec.bird, data.env.bird, xy.bird, sam = 100, order = 3,
  reg.type = "ngls", rescale = TRUE)
zeta.ngls

#####

utils::data(Marion.species)
xy.marion <- Marion.species[1:2]
data.spec.marion <- Marion.species[3:33]
utils::data(Marion.env)
data.env.marion <- Marion.env[3]

zeta.gam <- Zeta.msgdm(data.spec.marion, data.env.marion, sam = 100, order = 3,
  reg.type = "gam")
zeta.gam
dev.new()
graphics::plot(zeta.gam$model)

zeta.ispline <- Zeta.msgdm(data.spec.marion, data.env.marion, xy.marion, sam = 100,
  order = 3, normalize = "Jaccard", reg.type = "ispline")
zeta.ispline

zeta.ispline.r <- Return.ispline(zeta.ispline, data.env.marion, distance = TRUE)
zeta.ispline.r

dev.new()
Plot.ispline(isplines = zeta.ispline.r, distance = TRUE)

dev.new()
Plot.ispline(msgdm = zeta.ispline, data.env = data.env.marion, distance = TRUE)

```

Description

Computes the expectation of zeta diversity, the number of species shared by multiple assemblages, for a specific order (number of assemblages or sites) using a formula based on the occupancy of the species.

Usage

```
Zeta.order.ex(
  data.spec,
  order = 1,
  sd.correct = TRUE,
  rescale = FALSE,
  empty.row = "empty"
)
```

Arguments

data.spec	Site-by-species presence-absence data frame, with sites as rows and species as columns.
order	Specific number of assemblages or sites at which zeta diversity is computed.
sd.correct	Boolean value (TRUE or FALSE) indicating if the standard deviation must be computed with an unbiased estimator (using the number of site combinations - 1 as the denominator) or not (using the number of site combinations as the denominator).
rescale	Boolean value (TRUE or FALSE) indicating if the zeta values should be divided by ζ_1 , to get a range of values between 0 and 1.
empty.row	Determines how to handle empty rows, i.e. sites with no species. Such sites can cause underestimations of zeta diversity. Options are "empty" to let the data untreated or "remove" to remove the empty rows.

Details

Zeta.order.ex is much faster than [Zeta.order.mc](#) to compute the exact value of zeta diversity when the number of species is lower than C_i^N , where N is the total number of sites and i is the order of zeta.

sd.correct should be set to TRUE if the assemblages represent a subsample of the whole system. It can be set to FALSE if the sampling is exhaustive, for example in case of a continuous regular grid covering the whole study area.

Value

zeta.order.ex returns a list containing the following components:

zeta.order	The number of assemblages or sites for which the zeta diversity was computed.
combinations	The number of possible combinations of sites for the chosen order.
zeta.val	The zeta diversity values.
zeta.val.sd	The standard deviation of zeta diversity.

References

Hui C. & McGeoch M.A. (2014). zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

McGeoch M. A., Latombe G., Andrew N. R., Nakagawa S., Nipperess D. A., Roige M., Marzinelli E. M., Campbell A. H., Verges A., Thomas T., Steinberg P. D., Selwood K. E., Henriksen M. V. & Hui C. (2019). Measuring continuous compositional change using decline and decay in zeta diversity. *Ecology*, 100(11), e02832.

See Also

[Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.decline.mc](#)

Examples

```
utils::data(bird.spec.coarse)
data.spec.bird <- bird.spec.coarse[,3:193]

zeta.bird <- Zeta.order.ex(data.spec.bird, order = 3)
zeta.bird

#####
s
utils::data(Marion.species)
data.spec.marion <- Marion.species[,3:33]

zeta.marion <- Zeta.order.ex(data.spec.marion, order = 3)
zeta.marion
```

Zeta.order.mc	<i>Zeta diversity for a specific number of assemblages or sites using Monte Carlo sampling</i>
---------------	--

Description

Computes zeta diversity, the number of species shared by multiple assemblages, for a specific order (number of assemblages or sites).

Usage

```
Zeta.order.mc(
  data.spec,
  xy = NULL,
  order = 1,
  sam = 1000,
  sd.correct = TRUE,
  sd.correct.adapt = FALSE,
```

```

    rescale = FALSE,
    normalize = FALSE,
    NON = FALSE,
    FPO = NULL,
    DIR = FALSE,
    empty.row = "empty",
    silent = TRUE
)

```

Arguments

<code>data.spec</code>	Site-by-species presence-absence data frame, with sites as rows and species as columns.
<code>xy</code>	Site coordinates. This is only used if <code>NON = TRUE</code> or <code>DIR = TRUE</code> .
<code>order</code>	Specific number of assemblages or sites at which zeta diversity is computed.
<code>sam</code>	Number of samples for which the zeta diversity is computed.
<code>sd.correct</code>	Boolean value (TRUE or FALSE) indicating if the standard deviation must be computed with an unbiased estimator (using the number of site combinations - 1 as the denominator) or not (using the number of site combinations as the denominator).
<code>sd.correct.adapt</code>	Boolean value (TRUE or FALSE) indicating if the standard deviation must be computed with an unbiased estimator (using the number of site combinations - 1 as the denominator) if <code>sam</code> is higher than the number of possible combinations, or not (using the number of site combinations as the denominator) if <code>sam</code> is lower than the number of possible combinations. If <code>sd.correct.adapt == TRUE</code> , it takes precedence over <code>sd.correct</code> .
<code>rescale</code>	Boolean value (TRUE or FALSE) indicating if the zeta values should be divided by ζ_1 , to get a range of values between 0 and 1.
<code>normalize</code>	Indicates if the zeta values for each sample should be divided by the total number of species for this specific sample (<code>normalize = "Jaccard"</code>), by the average number of species per site for this specific sample (<code>normalize = "Sorensen"</code>), or by the minimum number of species in the sites of this specific sample (<code>normalize = "Simpson"</code>). Default value is FALSE, indicating that no normalization is performed.
<code>NON</code>	Boolean value (TRUE or FALSE) indicating if the number of species in common should only be counted for the nearest neighbours.
<code>FPO</code>	A vector with the coordinates of the fixed point origin from which the zeta diversity will be computed (overrides <code>NON</code>). In that case, ζ_1 is the number of species in the closest site to the FPO, ζ_2 is the number of species shared by the 2 closest sites, etc.
<code>DIR</code>	Boolean value (TRUE or FALSE) indicating if zeta diversity must be computed using a directed nearest neighbour scheme in the direction away from the FPO, starting from any site.

empty.row	Determines how to handle empty rows, i.e. sites with no species. Such sites can cause underestimations of zeta diversity, and computation errors for the normalized version of zeta due to divisions by 0. Options are "empty" to let the data untreated, "remove" to remove the empty rows, 0 to set the normalized zeta to 0 when zeta is divided by 0 during normalization (sites share no species, so are completely dissimilar), and 1 to set the normalized zeta to 1 when zeta is divided by 0 during normalization (i.e. sites are perfectly similar).
silent	Boolean value (TRUE or FALSE) indicating if messages must be printed.

Details

If the number of combinations of sites is lower than the value of the parameter `sam`, all the combinations are used and an exact solution is computed. In that case, using the number of site combinations as the denominator may be appropriate to compute the standard deviation, if all sites were sampled and the zeta values. This can be adjusted with parameters `sd.correct` and `sd.correct.adapt`.

`Zeta.order.mc` is faster than [Zeta.order.ex](#) to compute the exact value of zeta diversity when the number of species is higher than C_i^N , where N is the total number of sites and i is the order of zeta.

`Zeta.order.mc` enables accomodating richness heterogeneity by setting `normalize = "Jaccard"`, `normalize = "Sorensen"` or `normalize = "Simpson"`. This cannot be performed by [Zeta.order.ex](#).

Value

`Zeta.order.mc` returns a list containing the following components:

<code>zeta.order</code>	The number of assemblages or sites for which the zeta diversity was computed.
<code>combinations</code>	The number of possible combinations of sites for the chosen order.
<code>zeta.val</code>	The zeta diversity values.
<code>zeta.val.sd</code>	The standard deviation of zeta diversity.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

See Also

[Zeta.decline.mc](#)

Examples

```
utils::data(bird.spec.coarse)
xy.bird <- bird.spec.coarse[,1:2]
data.spec.bird <- bird.spec.coarse[,3:193]

zeta.bird <- Zeta.order.mc(data.spec.bird, order = 3, sam=100)
```



```

zeta.bird

#####

utils::data(Marion.species)
xy.marion <- Marion.species[,1:2]
data.spec.marion <- Marion.species[,3:33]

zeta.marion <- Zeta.order.mc(data.spec.marion, xy.marion, order = 3, sam = 100,
  NON = TRUE)
zeta.marion

```

Zeta.order.mc.mult	<i>Number of species in common between a specific number of assemblages or sites using Monte Carlo sampling, for multiple combinations and several groups of taxa</i>
--------------------	---

Description

Computes the number of species shared by multiple assemblages, for a specific order (number of assemblages or sites), for multiple combinations and several groups of taxa.

Usage

```

Zeta.order.mc.mult(
  data.spec,
  xy = NULL,
  order = 1,
  sam = 1000,
  sd.correct = TRUE,
  sd.correct.adapt = FALSE,
  rescale = FALSE,
  normalize = FALSE,
  NON = FALSE,
  FPO = NULL,
  DIR = FALSE,
  empty.row = "empty",
  silent = TRUE
)

```

Arguments

data.spec	A list of site-by-species presence-absence data frames, with sites as rows and species as columns.
xy	Site coordinates. This is only used if NON = TRUE or DIR = TRUE.
order	Specific number of assemblages or sites at which zeta diversity is computed.

<code>sam</code>	Number of samples for which the zeta diversity is computed.
<code>sd.correct</code>	Boolean value (TRUE or FALSE) indicating if the standard deviation must be computed with an unbiased estimator (using the number of site combinations - 1 as the denominator) or not (using the number of site combinations as the denominator).
<code>sd.correct.adapt</code>	Boolean value (TRUE or FALSE) indicating if the standard deviation must be computed with an unbiased estimator (using the number of site combinations - 1 as the denominator) if <code>sam</code> is higher than the number of possible combinations, or not (using the number of site combinations as the denominator) if <code>sam</code> is lower than the number of possible combinations. If <code>sd.correct.adapt == TRUE</code> , it takes precedence over <code>sd.correct</code> .
<code>rescale</code>	Boolean value (TRUE or FALSE) indicating if the zeta values should be divided by ζ_1 , to get a range of values between 0 and 1.
<code>normalize</code>	Indicates if the zeta values for each sample should be divided by the total number of species for this specific sample (<code>normalize = "Jaccard"</code>), by the average number of species per site for this specific sample (<code>normalize = "Sorensen"</code>), or by the minimum number of species in the sites of this specific sample (<code>normalize = "Simpson"</code>). Default value is FALSE, indicating that no normalization is performed.
<code>NON</code>	Boolean value (TRUE or FALSE) indicating if the number of species in common should only be counted for the nearest neighbours.
<code>FPO</code>	A vector with the coordinates of the fixed point origin from which the zeta diversity will be computed (overrides <code>NON</code>). In that case, ζ_1 is the number of species in the closest site to the FPO, ζ_2 is the number of species shared by the 2 closest sites, etc.
<code>DIR</code>	Boolean value (TRUE or FALSE) indicating if zeta diversity must be computed using a directed nearest neighbour scheme in the direction away from the FPO, starting from any site.
<code>empty.row</code>	Determines how to handle empty rows, i.e. sites with no species. Such sites can cause underestimations of zeta diversity, and computation errors for the normalized version of zeta due to divisions by 0. Options are "empty" to let the data untreated, "remove" to remove the empty rows, 0 to set the normalized zeta to 0 when zeta is divided by 0 during normalization (sites share no species, so are completely dissimilar), and 1 to set the normalized zeta to 1 when zeta is divided by 0 during normalization (i.e. sites are perfectly similar).
<code>silent</code>	Boolean value (TRUE or FALSE) indicating if messages must be printed.

Details

Contrary to `Zeta.order.mc`, the number of species shared by the different combinations of assemblages are not averaged, but returned as is. This is useful to then compare local zeta diversity for different groups of taxa.

As for `Zeta.order.mc`, if the number of combinations of sites is lower than the value of the parameter `sam`, all the combinations are used and an exact solution is computed. In that case, using the

number of site combinations as the denominator may be appropriate to compute the standard deviation, if all sites were sampled and the zeta values. This can be adjusted with parameters `sd.correct` and `sd.correct.adapt`.

Value

`Zeta.order.mc.mult` returns a list containing the following components:

<code>zeta.order</code>	The number of assemblages or sites for which the zeta diversity was computed.
<code>#' @return</code>	
<code>sites</code>	A matrix in which each row contains the indices of a given combination, i.e. of the specific sam assemblages.
<code>zeta.val</code>	A data frame in which each column is the number of species shared by the assemblages.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

See Also

[Zeta.decline.mc](#)

Examples

```
utils::data(Marion.species)
xy.marion <- Marion.species[1:2]

data.spec.marion <- Marion.species[3:33]

##random other communities
data.spec.marion2a <- data.spec.marion
data.spec.marion2a[which(data.spec.marion2a==1,arr.ind=TRUE)] <- 0
for(i in 1:ncol(data.spec.marion2a))
  data.spec.marion2a[sample(nrow(data.spec.marion2a),8),i] <- 1
data.spec.marion2b <- data.spec.marion
data.spec.marion2b[which(data.spec.marion2b==1,arr.ind=TRUE)] <- 0
for(i in 1:ncol(data.spec.marion2b))
  data.spec.marion2b[sample(nrow(data.spec.marion2b),8),i] <- 1

dat.spec.tot <- list(data.spec.marion,data.spec.marion2a,data.spec.marion2b)
zeta.tot <- Zeta.order.mc.mult(data.spec=dat.spec.tot,order=3,sam=200)
```

Zeta.sam.sensitivity *Sensitivity analysis for the sample size of zeta*

Description

Computes zeta diversity for a given order (number of assemblages or sites) for a range of sample sizes, to assess the sensitivity to this parameter.

Usage

```
Zeta.sam.sensitivity(
  data.spec,
  xy = NULL,
  order = 1,
  sam.seq,
  reps = 20,
  sd.correct = TRUE,
  sd.correct.adapt = FALSE,
  rescale = FALSE,
  normalize = FALSE,
  NON = FALSE,
  FPO = NULL,
  DIR = FALSE,
  display = TRUE,
  plot = TRUE,
  notch = TRUE
)
```

Arguments

data.spec	Site-by-species presence-absence data frame, with sites as rows and species as columns.
xy	Site coordinates. This is only used if NON = TRUE or DIR = TRUE.
order	Specific number of assemblages or sites at which zeta diversity is computed.
sam.seq	Sequence of samples for which the zeta diversity is computed.
reps	Number of replicates of zeta diversity computations for each sample size.
sd.correct	Boolean value (TRUE or FALSE) indicating if the standard deviation must be computed with an unbiased estimator (using the number of site combinations - 1 as the denominator) or not (using the number of site combinations as the denominator).
sd.correct.adapt	Boolean value (TRUE or FALSE) indicating if the standard deviation must be computed with an unbiased estimator (using the number of site combinations - 1 as the denominator) if sam is higher than the number of possible combinations, or not (using the number of site combinations as the denominator) if sam is lower

	than the number of possible combinations. If <code>sd.correct.adapt == TRUE</code> , it takes precedence over <code>sd.correct</code> .
<code>rescale</code>	Boolean value (TRUE or FALSE) indicating if the zeta values should be divided by ζ_1 , to get a range of values between 0 and 1.
<code>normalize</code>	Indicates if the zeta values for each sample should be divided by the total number of species for this specific sample (<code>normalize = "Jaccard"</code>), by the average number of species per site for this specific sample (<code>normalize = "Sorensen"</code>), or by the minimum number of species in the sites of this specific sample (<code>normalize = "Simpson"</code>). Default value is FALSE, indicating that no normalization is performed.
<code>NON</code>	Boolean value (TRUE or FALSE) indicating if the number of species in common should only be counted for the nearest neighbours.
<code>FPO</code>	A vector with the coordinates of the fixed point origin from which the zeta diversity will be computed (overrides <code>NON</code>). In that case, ζ_1 is the number of species in the closest site to the FPO, ζ_2 is the number of species shared by the 2 closest sites, etc.
<code>DIR</code>	Boolean value (TRUE or FALSE) indicating if zeta diversity must be computed using a directed nearest neighbour scheme in the direction away from the FPO, starting from any site.
<code>display</code>	Boolean value (TRUE or FALSE) indicating if the current value of the sample size must be displayed. Acts as a counter.
<code>plot</code>	Boolean value (TRUE or FALSE) indicating if the outputs must be plotted as a boxplot of the zeta diversity distributions for each sample size
<code>notch</code>	Boolean value (TRUE or FALSE) indicating if the notches must be plotted in the boxplot.

Details

Note that the execution of `Zeta.sam.sensitivity` can be quite lengthy, because of the number of replicates needed.

Value

`Zeta.sam.sensitivity` returns a matrix with `(sam.max-sam.min)/sam.incr` columns and `reps` rows.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#)

Examples

#Note that the sensitivity analyses in the following two examples are quite long to run, #typically around 10 minutes for the first example and 1-2 minutes for the second.

```
utils::data(bird.spec.coarse)
xy.bird <- bird.spec.coarse[1:2]
data.spec.bird <- bird.spec.coarse[3:193]

dev.new()
zeta.sens.bird <- Zeta.sam.sensitivity(data.spec.bird, order = 3,
  sam.seq = seq(250,1000,250), reps = 20, display = TRUE, plot = TRUE, notch = TRUE)
zeta.sens.bird

#####

utils::data(Marion.species)
xy.marion <- Marion.species[1:2]
data.spec.marion <- Marion.species[3:33]

dev.new()
zeta.sens.marion <- Zeta.sam.sensitivity(data.spec.marion, order = 3,
  sam.seq = seq(50,250,50), reps = 20, plot = TRUE, notch = TRUE)
zeta.sens.marion
```

Zeta.scale.min.dist *Zeta diversity scaling with sample grain dependency based on the minimum distance between sites*

Description

Computes zeta diversity scaling with sample grain for a specific order (number of assemblages or sites), increasing grain by sequentially adding sites based on the minimum distance between them.

Usage

```
Zeta.scale.min.dist(
  xy,
  data.spec,
  m,
  order = 1,
  reorder = 100,
  shuffle = TRUE,
  sam = 1000,
  method = "mean",
  rescale = FALSE,
  normalize = FALSE,
```

```

plot = TRUE,
sd = TRUE,
distance.type = "Euclidean",
dist.custom = NULL,
zeta.type = "exact"
)

```

Arguments

<code>xy</code>	Site-by-coordinate data frame, with sites as rows and coordinates as columns.
<code>data.spec</code>	Site-by-species presence-absence data frame, with sites as rows and species as columns.
<code>m</code>	Vector of mapping grains: <code>m[i]</code> sites are grouped together to generate data at a coarser grain.
<code>order</code>	Specific number of assemblages or sites at which zeta diversity is computed.
<code>reorder</code>	Number of times the sites are rearranged and grouped together for the computation of zeta (see Details).
<code>shuffle</code>	Boolean value (TRUE or FALSE) indicating if the order of the sites must be randomised, which can have an impact on the outputs if some distances are equal.
<code>sam</code>	Number of samples for which the zeta diversity is computed.
<code>method</code>	Name of a function (as a string) indicating how to combine the coordinates. It can be a basic R-function such as "mean" or "max", but also a custom function.
<code>rescale</code>	Boolean value (TRUE or FALSE) indicating if the zeta values should be divided by ζ_1 , to get a range of values between 0 and 1. Has no effect if <code>normalize != FALSE</code> .
<code>normalize</code>	Indicates if the zeta values for each sample should be divided by the total number of species for this specific sample (<code>normalize = "Jaccard"</code>), by the average number of species per site for this specific sample (<code>normalize = "Sorensen"</code>), or by the minimum number of species in the sites of this specific sample (<code>normalize = "Simpson"</code>). Default value is FALSE, indicating that no normalization is performed.
<code>plot</code>	Boolean value (TRUE or FALSE) indicating if the outputs must be plotted.
<code>sd</code>	Boolean value (TRUE or FALSE) indicating if the standard deviation must be plotted for each grain.
<code>distance.type</code>	Method to compute distance. Default is "Euclidean", for Euclidean distance. The other options are (i) "ortho" for orthodromic distance, if <code>xy</code> correspond to longitudes and latitudes (orthodromic distance is computed using the <code>geodist</code> function from package <code>geodist</code>); and (ii) "custom", in which case the user must provide a distance matrix for <code>dist.custom</code> .
<code>dist.custom</code>	Distance matrix provided by the user when <code>distance.type = "custom"</code> .
<code>zeta.type</code>	The function that must be used for the computation of zeta diversity. Default is "exact" for calling <code>Zeta.order.ex</code> . Use "monte carlo" for calling <code>Zeta.order.mc</code> .

Details

The nearest neighbouring sites (plots, quadrates, or areas of varying shapes) are grouped as spatial clusters of 2, 3, 4, etc. sites, based on the minimum distance between them. Since the procedure is based on the relative distance between sites, the site order can have an impact on the output. The procedure is therefore performed 'reorder' times, for which sites are randomly reordered each time, and the mean zeta is computed. This function is suitable for both regularly and irregularly spaced sites, contiguous or non contiguous (*sensu* Scheiner et al., 2011). For regularly spaced sites, the use of [Zeta.scale.regular](#) is recommended.

Value

zeta.scale.min.dist returns a list containing the following components:

order	The order of zeta.
m	The vector of mapping grains: m[i] sites are grouped together to generate data at a coarser grain.
values	A matrix containing the zeta diversity values over the 'reorder' computations, for each grain.
sd	A matrix containing the standard deviation of zeta diversity over the 'reorder' computations, for each grain.

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

Scheiner S.M., Chiarucci A., Fox G.A., Helmus M.R., McGlenn D.J. & Willig M.R. (2011). The underpinnings of the relationship of species richness with space and time. *Ecological Monographs*, 81, 195-213.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#),
[Zeta.scale.regular](#), [rescale.regular](#)

Examples

```
utils::data(Marion.species)
xy.marion <- Marion.species[,1:2]
data.spec.marion <- Marion.species[,3:33]

dev.new()
zeta.scale.irreg.species <- Zeta.scale.min.dist(xy.marion, data.spec.marion, m = 1:3,
  order = 3, reorder = 3, sam = 50, normalize = "Jaccard")
```

Zeta.scale.regular *Zeta diversity scaling with sample grain using hierarchical increases in grain size*

Description

Computes zeta diversity scaling with sample grain for a specific order (number of assemblages or sites), increasing grain by hierarchically nesting of regularly spaced sites.

Usage

```
Zeta.scale.regular(
  xy,
  data.spec,
  n,
  order = 1,
  sam = 1000,
  method = "mean",
  rescale = FALSE,
  normalize = FALSE,
  plot = TRUE,
  zeta.type = "exact"
)
```

Arguments

xy	Site-by-coordinate data frame, with sites as rows and coordinates as columns.
data.spec	Site-by-species presence-absence data frame, with sites as rows and species as columns.
n	Vector of mapping grains: regularly spaced sites are grouped as $n[i] \times n[i]$ sites to generate data at a coarser grain.
order	Specific number of assemblages or sites at which zeta diversity is computed.
sam	Number of samples for which the zeta diversity is computed.
method	Name of a function (as a string) indicating how to combine the coordinates. It can be a basic R-function such as "mean" or "max", but also a custom function.
rescale	Boolean value (TRUE or FALSE) indicating if the zeta values should be divided by ζ_1 , to get a range of values between 0 and 1. Has no effect if <code>normalize != FALSE</code> .
normalize	Indicates if the zeta values for each sample should be divided by the total number of species for this specific sample (<code>normalize = "Jaccard"</code>), by the average number of species per site for this specific sample (<code>normalize = "Sorensen"</code>), or by the minimum number of species in the sites of this specific sample (<code>normalize = "Simpson"</code>). Default value is FALSE, indicating that no normalization is performed.
plot	Boolean value (TRUE or FALSE) indicating if the outputs must be plotted.

`zeta.type` The function that must be used for the computation of zeta diversity. Default is "exact" for calling `Zeta.order.ex`. Use "monte carlo" for calling `Zeta.order.mc`.

Details

The sites (plots or quadrates) are incrementally aggregated as nearest neighbouring groups of 4, 9, etc. sites, using a nested approach, starting from the lowest x and y, to increase the grain. The sites can be spatially contiguous or discontinuous, as long as they are regularly spaced (see Scheiner et al., 2011). If the total number of sites is not a multiple of $n[i] \times n[i]$, the extra sites are discarded.

Value

`Zeta.scale.regular` returns a list containing the following components:

<code>order</code>	The order of zeta.
<code>n</code>	The vector of mapping grains: regularly spaced sites are grouped as $n[i] \times n[i]$ sites to generate data at a coarser grain.
<code>values</code>	The zeta diversity values for each grain.
<code>sd</code>	The standard deviation of zeta diversity for each grain.

References

- Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.
- Scheiner S.M., Chiarucci A., Fox G.A., Helmus M.R., McGlenn D.J. & Willig M.R. (2011). The underpinnings of the relationship of species richness with space and time. *Ecological Monographs*, 81, 195-213.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#)
[Zeta.scale.min.dist](#), [rescale.regular](#), [rescale.min.dist](#)

Examples

```
utils::data(bird.spec.fine)
xy.bird <- bird.spec.fine[1:400,1:2]
data.spec.bird <- bird.spec.fine[1:400,3:192]

dev.new()
##sam = 25 is used here for fast execution, but a higher value is advised
zeta.scale.reg <- Zeta.scale.regular(xy.bird, data.spec.bird, n = 1:3, order = 3,
  sam = 25, normalize = "Jaccard", zeta.type="monte carlo")
```

Zeta.varpart *Variation partitioning for zeta diversity*

Description

Variation partitioning of zeta diversity for a specific order (number of assemblages or sites) over distance and environmental variables.

Usage

```
Zeta.varpart(
  msgdm.mod,
  num.part = 2,
  reg.type = "glm",
  family = stats::gaussian(),
  method.glm = "glm.fit.cons",
  cons = -1,
  cons.inter = 1,
  kn = -1,
  bs = "mpd"
)
```

Arguments

msgdm.mod	An object return by function Zeta.msgdm .
num.part	Number of partitions of zeta diversity. Can be 2 or 3.
reg.type	Type of regression for the multi-site generalised dissimilarity modelling. Options are "glm" for generalised linear models, "ngls" for negative linear models, "gam" for generalised additive models, "scam" for shape constrained additive models, and "ispline" for I-spline models, as recommended in generalised dissimilarity modelling by Ferrier <i>et al.</i> (2007).
family	A description of the error distribution and link function to be used in the glm, gam and scam models (see family for details of family functions).
method.glm	Method used in fitting the generalised linear model. The default method "glm.fit.cons" is an adaptation of method <code>glm.fit2</code> from package <code>glm2</code> using a negative least squares regression in the reweighted least squares. Another option is "glm.fit2", which corresponds to method <code>glm.fit2</code> ; see help documentation for <code>glm.fit2</code> in package <code>glm</code> .
cons	type of constraint in the glm if <code>method.glm = "glm.fit.cons"</code> . Default is -1 for negative coefficients on the predictors. The other option is 1 for positive coefficients on the predictors.
cons.inter	type of constraint for the intercept. Default is 1 for positive intercept, suitable for Gaussian family. The other option is -1 for negative intercept, suitable for binomial family.

kn	Number of knots in the GAM and SCAM. Default is -1 for determining kn automatically using Generalized Cross-validation.
bs	A two-letter character string indicating the (penalized) smoothing basis to use in the scam model. Default is "mpd" for monotonic decreasing splines. see smooth.terms for an overview of what is available.

Details

Note that, for a given regression, the variation explained is computed as $1 - (\text{RSS}/\text{TSS})^{(v-1)/(v-p-1)}$, where RSS is the residual sum of squares and TSS is the total sum of squares, v is the number of variables used in the regression (which is greater than the original number of variables for I-splines) and p is the number of samples. $1 - (\text{RSS}/\text{TSS})$ corresponds to the classical R-squared for linear regression only, and results for non-linear regressions should be interpreted with caution.

The environmental variables can be numeric or factorial, and order must be greater than 1.

For numeric variables, the pairwise difference between sites is computed and combined according to method. For factorial variables, the distance corresponds to the number of unique values over the number of assemblages of sites specified by order.

Zeta is regressed against the differences of values of the environmental variables divided by the maximum difference for each variable, to be rescaled between 0 and 1. If `!is.null(xy)`, distances between sites are also divided by the maximum distance.

Value

Zeta.varpart returns a data frame with one column containing the variation explained by each component a (the variation explained by distance alone), b (the variation explained by either distance or the environment), c (the variation explained by the environment alone) and d (the unexplained variation).

References

Hui C. & McGeoch M.A. (2014). Zeta diversity as a concept and metric that unifies incidence-based biodiversity patterns. *The American Naturalist*, 184, 684-694.

Borcard, D., Legendre, P. & Drapeau, P. (1992). Partialling out the spatial component of ecological variation. *Ecology* 73, 1045-1055.

Legendre, P. & Legendre, L.F. (2012). *Numerical ecology*, 3rd English edition. Elsevier Science BV, Amsterdam.

See Also

[Zeta.decline.mc](#), [Zeta.order.mc](#), [Zeta.decline.ex](#), [Zeta.order.ex](#), [Zeta.msgdm](#), [pie.neg](#)

Examples

```
utils::data(bird.spec.coarse)
xy.bird <- bird.spec.coarse[,1:2]
data.spec.bird <- bird.spec.coarse[,3:193]
utils::data(bird.env.coarse)
data.env.bird <- bird.env.coarse[,3:9]
```

```
zeta.bird <- Zeta.msgdm(data.spec.bird, data.env.bird, xy.bird, sam = 100, order = 3)
zeta.varpart.bird <- Zeta.varpart(zeta.bird, method.glm = "glm.fit2")
zeta.varpart.bird
dev.new()
pie.neg(zeta.varpart.bird[4:7,1], density = c(4, 0, 8, -1),
        angle = c(90, 0, 0, 0),
        labels = c("distance", "undistinguishable", "environment", "unexplained"),
        radius = 0.9)

#####

utils::data(Marion.species)
xy.marion <- Marion.species[,1:2]
data.spec.marion <- Marion.species[,3:33]
utils::data(Marion.env)
data.env.marion <- Marion.env[3:4]

zeta.marion <- Zeta.msgdm(data.spec.marion, data.env.marion, xy.marion, sam = 100,
                        order = 3, normalize = "Jaccard")
zeta.varpart.marion <- Zeta.varpart(zeta.marion, method.glm = "glm.fit2")
zeta.varpart.marion
dev.new()
pie.neg(zeta.varpart.marion[4:7,1], density = c(4, 0, 8, -1),
        angle = c(90, 0, 0, 0),
        labels = c("distance", "undistinguishable", "environment", "unexplained"),
        radius = 0.9)
```

Index

* data

- bird.env.coarse, [2](#)
 - bird.env.fine, [3](#)
 - bird.spec.coarse, [4](#)
 - bird.spec.fine, [5](#)
 - Marion.env, [10](#)
 - Marion.species, [11](#)
- as.graphicsAnnot, [12](#)
- bird.env.coarse, [2](#)
- bird.env.fine, [3](#)
- bird.spec.coarse, [4](#)
- bird.spec.fine, [5](#)
- family, [23](#), [30](#), [33](#), [41](#), [59](#)
- glm, [6](#), [7](#), [23](#), [24](#), [42](#)
- glm.cons, [6](#)
- glm.fit, [8](#), [9](#)
- glm.fit.cons, [8](#)
- glm.fit2, [9](#)
- glm2, [7](#)
- Ispline, [9](#), [21](#), [24](#), [29](#), [43](#)
- Marion.env, [10](#)
- Marion.species, [11](#)
- pie, [13](#)
- pie.neg, [12](#), [60](#)
- Plot.ispline, [13](#)
- Plot.zeta.ddecay, [15](#), [17](#), [32](#)
- Plot.zeta.ddecays, [16](#)
- Plot.zeta.decline, [17](#), [36](#), [39](#)
- Plot.zeta.scale.min.dist, [18](#), [20](#)
- Plot.zeta.scale.regular, [19](#), [20](#)
- Predict.msgdm, [21](#), [43](#)
- Reg.ispline, [23](#)
- rescale.min.dist, [25](#), [28](#), [58](#)
- rescale.regular, [19](#), [20](#), [26](#), [27](#), [56](#), [58](#)
- Return.ispline, [28](#)
- smooth.terms, [31](#), [41](#), [60](#)
- Zeta.ddecay, [16](#), [17](#), [29](#), [34](#)
- Zeta.ddecays, [16](#), [17](#), [32](#), [32](#)
- Zeta.decline.ex, [16–20](#), [26](#), [28](#), [32](#), [34](#), [34](#), [38](#), [39](#), [43](#), [46](#), [53](#), [56](#), [58](#), [60](#)
- Zeta.decline.mc, [16–20](#), [26](#), [28](#), [32](#), [34–36](#), [37](#), [43](#), [46](#), [48](#), [51](#), [53](#), [56](#), [58](#), [60](#)
- Zeta.msgdm, [10](#), [15](#), [22](#), [24](#), [29](#), [40](#), [59](#), [60](#)
- Zeta.order.ex, [16–20](#), [26](#), [28](#), [32](#), [34](#), [36](#), [39](#), [43](#), [44](#), [48](#), [53](#), [56](#), [58](#), [60](#)
- Zeta.order.mc, [16–20](#), [26](#), [28](#), [32](#), [34](#), [36](#), [39](#), [43](#), [45](#), [46](#), [46](#), [53](#), [56](#), [58](#), [60](#)
- Zeta.order.mc.mult, [49](#)
- Zeta.sam.sensitivity, [52](#)
- Zeta.scale.min.dist, [19](#), [20](#), [26](#), [28](#), [54](#), [58](#)
- Zeta.scale.regular, [19](#), [20](#), [26](#), [28](#), [56](#), [57](#)
- Zeta.varpart, [13](#), [59](#)