Package 'ecodist'

July 2, 2014

Version 1.2.9

Date 2013-12-03

Title Dissimilarity-based functions for ecological analysis

Author Sarah Goslee and Dean Urban

Maintainer Sarah Goslee <Sarah.Goslee@ars.usda.gov>

Depends R (>= 3.0.0)

Imports stats

Description Dissimilarity-based analysis functions including ordination and Mantel test functions, intended for use with spatial and community data.

License GPL (>= 2)

NeedsCompilation yes

Repository CRAN

Date/Publication 2013-12-03 22:07:08

R topics documented:

ecodist-package 2
bcdist 5
bump 6
cor2m 6
corgen 7
crosstab 8
distance 10
fixdmat 11
full 11
graze 12
lower 15
mantel 15

mgram 17
MRM 19
nmds 20
nmds.min 22
pco 23
permresults 24
plot.mgram 26
plot.vf 27
pmgram 28
residuals.mgram 30
vf 31

Index **33**

`ecodist-package` *Dissimilarity-based functions for ecological analysis*

Description

Dissimilarity-based analysis functions including ordination and Mantel test functions with multiple partials, intended for use with spatial and community data.

Details

Package:	ecodist
Version:	1.2.2
Date:	2011-04-02
Depends:	stats
License:	GPL version 2 or newer

Index:

`bcdist`	`Bray-Curtis distance`
`cor2m`	`Generates a correlation table between the variables of 2 matrices`
`corgen`	`Generate correlated data`
`crosstab`	`Data formatting`
`distance`	`Calculate dissimilarity/distance metrics`
`fixdmat`	`Distance matrix conversion`
`full`	`Full symmetric matrix`
`lower`	`Lower-triangular matrix`
`mantel`	`Mantel test`
`mgram`	`Mantel correlogram`
`nmds`	`Non-metric multidimensional scaling`
`nmds.min`	`Find minimum stress configuration`
`pco`	`Principal coordinates analysis`

plot.mgram	Plot a Mantel correlogram
plot.vf	Plot fitted vectors onto an ordination diagram
pmgram	Partial Mantel correlogram
vf	Vector fitting

Author(s)

Sarah Goslee and Dean Urban

Maintainer: Sarah Goslee <Sarah.Goslee@ars.usda.gov>

References

Goslee, S.C. and Urban, D.L. 2007. The ecodist package for dissimilarity-based analysis of ecological data. Journal of Statistical Software 22(7):1-19.

Examples

```
## Distance example using a subset of the built-in iris dataset
data(iris)
iris <- iris[seq(1, 150, by=3),]
iris.md <- distance(iris[,1:4], "mahal")
iris.bc <- bcdist(iris[,1:4])

# compare Mahalanobis and Bray-Curtis dissimilarities
plot(iris.md, iris.bc, xlab="Mahalanobis", ylab="Bray-Curtis")


## NMDS example
# Example of multivariate analysis using built-in iris dataset

# Minimum-stress 2-dimensional nonmetric multidimensional scaling configuration
# Uses only 3 starting configuration to increase speed of example.
# Use more for final analysis to make it more likely that you find the
# global minimum configuration.
iris.nmds <- nmds(iris.md, mindim=2, maxdim=2, nits=3)
iris.nmin <- nmds.min(iris.nmds)

# Plot NMDS result with symbols denoting species
plot(iris.nmin, pch=as.numeric(iris[,5]))

# Fit vectors for the main variables to the NMDS configuration
# Use more permutations for analysis and publication.
iris.vf <- vf(iris.nmin, iris[,1:4], nperm=50)
plot(iris.vf, col="blue")


## PCO example

iris.pco <- pco(iris.md)
# scatterplot of the first two dimensions
plot(iris.pco$vectors[,1], iris.pco$vectors[,2], pch=as.numeric(iris[,5]))
```

```
## Mantel test example

# Example of multivariate analysis using built-in iris dataset

# Create a model matrix for testing species differences
iris.model <- distance(as.numeric(iris[,5]), "eucl")
iris.model[iris.model > 0] <- 1

# Test whether samples within the same species are more similar than those not
# Use very low numbers of permutations to increase speed of example.
# Use many permutations for final analyses!
mantel(iris.md ~ iris.model, nperm=50, nboot=0)


## Mantel correlogram example

# generate a simple surface
x <- matrix(1:10, nrow=10, ncol=10, byrow=FALSE)
y <- matrix(1:10, nrow=10, ncol=10, byrow=TRUE)
z <- x + 3*y
image(z)

# analyze the pattern of z across space
space <- cbind(as.vector(x), as.vector(y))
z <- as.vector(z)
space.d <- distance(space, "eucl")
z.d <- distance(z, "eucl")
z.mgram <- mgram(z.d, space.d, nperm=0)
plot(z.mgram)


## Partial Mantel correlogram example
# generate a simple surface
x <- matrix(1:10, nrow=10, ncol=10, byrow=FALSE)
y <- matrix(1:10, nrow=10, ncol=10, byrow=TRUE)
z1 <- x + 3*y
z2 <- 2*x - y

# look at patterns
par(mfrow=c(1,2))
image(z1)
image(z2)

# analyze the pattern of z across space
z1 <- as.vector(z1)
z2 <- as.vector(z2)
z1.d <- distance(z1, "eucl")
z2.d <- distance(z2, "eucl")

space <- cbind(as.vector(x), as.vector(y))
space.d <- distance(space, "eucl")
```

```
# take partial correlogram of z2 on the residuals of z1 ~ space.d
z.pmgram <- pmgram(z1.d, space.d, z2.d, nperm=0)
par(mfrow=c(1,1))
plot(z.pmgram, pval=0.1)
```

---

bcdist *Bray-Curtis distance*

Description

Returns the Bray-Curtis (also known as Sorenson, 1 - percent similarity) pairwise distances for the objects in the data. It has been superseded by distance() but remains for backward compatibility.

Usage

```
bcdist(x, rmzero = FALSE)
```

Arguments

x	matrix or data frame with rows as samples and columns as variables (such as species). Distances will be calculated for each pair of rows.
rmzero	If rzero=TRUE, empty rows will be removed from the data before distances are calculated. Otherwise, the distance between two empty rows is assumed to be 0 (the default).

Value

This function returns a column-order lower-triangular distance matrix. The returned object has an attribute, Size, giving the number of objects, that is, nrow(x). The length of the vector that is returned is nrow(x)*(nrow(x)-1)/2.

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

See Also

dist

Examples

```
data(iris)
iris.bc <- bcdist(iris[,1:4])
# equivalent to
iris.bc2 <- distance(iris[,1:4], "bray-curtis")
```

bump *Nine-bump spatial pattern*

Description

A two-dimensional artificial "landscape" illustrating the kind of spatial pattern that might be seen across mountain peaks.

Usage

```
data(bump)
```

Format

The format is: int [1:25, 1:25] 2 2 2 2 2 2 2 2 2 2 ... - attr(*, "dimnames")=List of 2 ..$: chr [1:25] "1" "3" "5" "7" ... ..$: chr [1:25] "V1" "V3" "V5" "V7" ...

Examples

```
data(bump)
image(bump)
## maybe str(bump) ; plot(bump) ...
```

cor2m *Generates a correlation table between the variables of 2 matrices*

Description

cor2m generates a correlation table between the variables of 2 matrices–by design, species and environment. It stores these in a table with species as columns and envnvironmental variables as rows, so it's easy to scan. It also removes correlations less than a user-specified alpha (0.05 by default)

Usage

```
cor2m(x, y, trim = TRUE, alpha = 0.05)
```

Arguments

x	A matrix of species (or other) variables
y	A matrix of environmental (or other) variables matching the sites of x
trim	If trim is TRUE, set rho<critical value(alpha) to 0
alpha	alpha p-value to use with trim

Details

Correlate variables from 1 matrix with variables from another matrix. If TRIM, set rho<critical value(alpha) to 0. Computes this critical value as a t-test with n-2 df.

Value

Returns a correlation table between the variables of 2 matrices.

Author(s)

Dean Urban

Examples

```
# toy example
speciesdata <- matrix(runif(50), nrow=10)
colnames(speciesdata) <- c("spA", "spB", "spC", "spD", "spE")
envdata <- matrix(runif(30), nrow=10)
colnames(envdata) <- c("var1", "var2", "var3")
sppenv.cor <- cor2m(speciesdata, envdata)
```

---

corgen *Generate correlated data*

Description

Generate correlated data vectors of a given length.

Usage

```
corgen(len, x, r, population = FALSE, epsilon = 0)
```

Arguments

len	Length of data vectors.
x	Independent data. If x is specified, the population parameter is automatically set to TRUE.
r	Correlation between data vectors.
population	TRUE for vectors drawn from two populations with correlation r, otherwise r is the sample correlation.
epsilon	If epsilon = 0, it has no effect, otherwise the sampling process is repeated until the sample correlation is within epsilon of r. This option allows the production of exactly-correlated data, within the limits of epsilon. Setting epsilon > 0 invalidates the population setting; data will be correlated within that range, rather than sampled from that population.

Details

Either x or len must be specified. This function has additional capabilities beyond the corgen function available in the stats package.

Value

x	First data vector, either generated by corgen or given by the user.
y	Second data vector.
...	

Author(s)

Sarah Goslee

Examples

```
# create two random variables of length 100 with correlation
# of 0.10 +/- 0.01
xy <- corgen(len=100, r=.1, epsilon=0.01)

# create two random variables of length 100 drawn from a population with
# a correlation of -0.82
xy <- corgen(len=100, r=-0.82, population=TRUE)

# create a variable y within 0.01 of the given correlation to x
x <- 1:100
y <- corgen(x=x, r=.5, epsilon=.01)$y
```

---

crosstab *Data formatting*

Description

Converts field data of the form site, species, observation into a matrix.

Usage

```
crosstab(rowlab, collab, values, type = "sum", data, allrows, allcols, ...)
```

Arguments

rowlab	row labels, e.g. site names.
collab	column labels, e.g. species names.
values	data values.
data	optional data argument to take rowlab, collab and/or values from.

type	function to use to combine data, one of "sum" (default), "min", "max", "mean", "count".
allrows	optional, list of all desired row names that may not appear in rowlab.
allcols	optional, list of all desired column names that may not appear in collab.
...	optional arguments to the function specified in type, such as na.rm=TRUE

Details

Field data are often recorded as a separate row for each site-species combination. This function turns them into a matrix for further analysis based on unique row and column labels.

The three vectors should all be the same length (including duplicates). The three vectors may also be provided as names of columns in the data frame specified by the data argument.

If allrows or allcols exists, rows and/or columns of zeros are inserted for any elements of allrows/allcols not present in rowlab/collab.

If values is missing the number of occurrences of combinations of rowlab and collab will be returned. Thus, crosstab(rowlab, collab) is equivalent to table(rowlab, collab).

If type is "count", the unique combinations of rowlab, collab and values will be returned.

Value

matrix with rowlab as row headings, collab as columns, and values as the data.

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

Examples

```
# Make a random example
plotnames <- sort(rep(1:5, 6))
speciesnames <- rep(c("A", "B", "C"), 10)
freqdata <- runif(30)

# number of samples of each species and plot
crosstab(plotnames, speciesnames)

# mean frequency by species and plot
crosstab(plotnames, speciesnames, freqdata, type="mean")
```

distance *Calculate dissimilarity/distance metrics*

Description

This function calculates a variety of dissimilarity or distance metrics. Although it duplicates the functionality of dist() and bcdist(), it is written in such a way that new metrics can easily be added. distance() was written for extensibility and understandability, and is not an efficient choice for use with large matrices.

Usage

```
distance(x, method = "euclidean", sprange=NULL, spweight=NULL)
```

Arguments

x	matrix or data frame with rows as samples and columns as variables (such as species). Distances will be calculated for each pair of rows.
method	Currently 7 dissimilarity metrics can be calculated: "euclidean", "bray-curtis", "manhattan", "mahalanobis", "jaccard", "difference", "sorensen", "gower", "modgower10" (modified Gower, base 10), "modgower2" (modified Gower, base 2). Partial matching will work for selecting a method.
sprange	Gower dissimilarities offer the option of dividing by the species range. If sprange=NULL no range is used. If sprange is a vector of length nrow(x) it is used for standardizing the dissimilarities.
spweight	Euclidean, Manhattan, and Gower dissimilarities allow weighting. If spweight=NULL, no weighting is used. If spweight="absence", then W=0 if both species are absent and 1 otherwise, thus deleting joint absences.

Value

Returns a lower-triangular distance matrix as an object of class "dist".

Author(s)

Sarah Goslee <Sarah.Goslee@ars.usda.gov>

See Also

dist

Examples

```
data(iris)
iris.md <- distance(iris[,1:4], "mahal")
```

fixdmat *Distance matrix conversion*

Description

Converts a row-order lower-triangular distance matrix to a full symmetric matrix.

Usage

```
fixdmat(v)
```

Arguments

v	lower-triangular distance matrix in row order.

Details

R and S-Plus distance functions such as dist and bcdist return a lower triangular distance matrix in column order. Some other programs, such as EUCLID, return the lower triangular matrix in row order. To use this matrix in R/S-Plus functions, it must be converted from row order to column order.

Value

Full symmetric distance matrix.

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

See Also

lower, full, dist, bcdist

full *Full symmetric matrix*

Description

Converts a column order distance matrix to a full symmetric matrix.

Usage

```
full(v)
```

Arguments

v	lower-triangular column order distance matrix.

Details

Converts lower-triangular distance matrix as written by R/S-Plus functions into a symmetric matrix. Note that lower() used on a 1x1 matrix will return the single element, which may not be the correct behavior in all cases, while full() used on a single element will return a 2x2 matrix.

Value

a full symmetric matrix.

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

See Also

lower

---

graze *Site information and grazed vegetation data.*

Description

These data contain site location, landscape context and dominant plant species abundances for 50 grazed pastures in the northeastern United States. Elements are the mean values for canopy cover for ten 0.5 x 2 m quadrats.

Usage

```
data(graze)
```

Format

A data frame with 50 observations on the following 67 variables.

`sitelocation` Site location along a geographic gradient.

`forestpct` Percentage forest cover within a 1-km radius.

`ACMI2` percentage canopy cover

`AGGI2` percentage canopy cover

`AGROS2` percentage canopy cover

`ANAR6` percentage canopy cover

`ANOD` percentage canopy cover

`ARMI2` percentage canopy cover

ASSY percentage canopy cover

BRCA5 percentage canopy cover

BRIN2 percentage canopy cover

CEFO2 percentage canopy cover

CIAR4 percentage canopy cover

CIIN percentage canopy cover

CIVU percentage canopy cover

DACA6 percentage canopy cover

DAGL percentage canopy cover

DRYOP percentage canopy cover

ELRE4 percentage canopy cover

ERAN percentage canopy cover

EUGRG percentage canopy cover

FERU2 percentage canopy cover

FRVI percentage canopy cover

GAMO percentage canopy cover

HIERA percentage canopy cover

HOLA percentage canopy cover

HYPU percentage canopy cover

HYRA3 percentage canopy cover

JUEF percentage canopy cover

LEAU2 percentage canopy cover

LEVU percentage canopy cover

LOAR10 percentage canopy cover

LOCO6 percentage canopy cover

LOPE percentage canopy cover

MOVE percentage canopy cover

OXST percentage canopy cover

PEGL2 percentage canopy cover

PHAR3 percentage canopy cover

PHPR3 percentage canopy cover

PLLA percentage canopy cover

PLMA2 percentage canopy cover

POAV percentage canopy cover

POCA17 percentage canopy cover

POLA4 percentage canopy cover

POLYG percentage canopy cover

PONO3 percentage canopy cover
POPR percentage canopy cover
POSI2 percentage canopy cover
POTEN percentage canopy cover
PRVU percentage canopy cover
RAAC3 percentage canopy cover
RUBUS percentage canopy cover
RUCR percentage canopy cover
SIOF percentage canopy cover
SOLID percentage canopy cover
SORU2 percentage canopy cover
STGR percentage canopy cover
STME2 percentage canopy cover
SYLA4 percentage canopy cover
TAOF percentage canopy cover
THVU percentage canopy cover
TRAU2 percentage canopy cover
TRHY percentage canopy cover
TRPR2 percentage canopy cover
TRRE3 percentage canopy cover
VEOF2 percentage canopy cover
VICIA percentage canopy cover

Details

Site locations fall along a southwest-northeast transect through the northeastern United States. This is a synthetic gradient calculated from latitude and longitude. Forest landcover is taken from the USGS 1992 National Land Cover Dataset. All forest classes were combined, and the percentage within 1 km of the sample sites was calculated using a GIS. Species codes follow the USDA Plants database. Species with a maximum cover <= 1 have been omitted.

Source

Details of these data are available in Tracy and Sanderson (2000) and Goslee, Sanderson and Tracy (ms in review). The 1992 NLCD data can be obtained from http://www.mrlc.gov/. Species codes are from http://plants.usda.gov.

References

Tracy, B.F. and M.A. Sanderson. 2000. Patterns of plant species richness in pasture lands of the northeast United States. Plant Ecology 149:169-180.

Examples

```
data(graze)
```

lower *Lower-triangular matrix*

Description

Converts a symmetric distance matrix to a column order lower triangular matrix.

Usage

```
lower(m)
```

Arguments

m	a symmetric distance matrix.

Details

Converts a symmetric matrix, for example a dissimilarity matrix, into a lower- triangular matrix. This may be useful to format the input for certain clustering and ordination functions. Note that lower() used on a 1x1 matrix will return the single element, which may not be the correct behavior in all cases, while full() used on a single element will return a 2x2 matrix.

Value

a column order lower triangular matrix.

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

See Also

full, fixdmat,

mantel *Mantel test*

Description

Simple and partial Mantel tests, with options for ranked data, permutation tests, and bootstrapped confidence limits.

Usage

```
mantel(formula = formula(data), data = sys.parent(), nperm = 1000,
    mrank = FALSE, nboot = 500, pboot = 0.9, cboot = 0.95)
```

Arguments

formula	formula in R/S-Plus format describing the test to be conducted. For this test, y ~ x will perform a simple Mantel test, while y ~ x + z1 + z2 + z3 will do a partial Mantel test of the relationship between x and y given z1, z2, z3. All variables can be either a distance matrix of class dist or vectors of dissimilarities.
data	an optional dataframe containing the variables in the model as columns of dissimilarities. By default the variables are taken from the current environment.
nperm	number of permutations to use. If set to 0, the permutation test will be omitted.
mrank	if this is set to FALSE (the default option), Pearson correlations will be used. If set to TRUE, the Spearman correlation (correlation ranked distances) will be used.
nboot	number of iterations to use for the bootstrapped confidence limits. If set to 0, the bootstrapping will be omitted.
pboot	the level at which to resample the data for the bootstrapping procedure.
cboot	the level of the confidence limits to estimate.

Details

If only one independent variable is given, the simple Mantel r (r12) is calculated. If more than one independent variable is given, the partial Mantel r (ryx|x1 ...) is calculated by permuting one of the original dissimilarity matrices. The bootstrapping is actually resampling without replacement, because duplication of samples is not useful in a dissimilarity context (the dissimilarity of a sample with itself is zero). Resampling within dissimilarity values is inappropriate, just as for permutation.

Value

mantelr	Mantel coefficient.
pval1	one-tailed p-value (null hypothesis: r <= 0).
pval2	one-tailed p-value (null hypothesis: r >= 0).
pval3	two-tailed p-value (null hypothesis: r = 0).
llim	lower confidence limit.
ulim	upper confidence limit.

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

References

Mantel, N. 1967. The detection of disease clustering and a generalized regressio n approach. Cancer Research 27:209-220.

Smouse, P.E., J.C. Long and R.R. Sokal. 1986. Multiple regression and correlatio n extensions of the Mantel test of matrix correspondence. Systematic Zoology 35:62 7-632.

See Also

mgram

Examples

```
## Not run:
# Example of multivariate analysis using built-in iris dataset
data(iris)
iris.md <- distance(iris[,1:4], "mahal")

# Create a model matrix for testing species differences
iris.model <- distance(as.numeric(iris[,5]), "eucl")
iris.model[iris.model > 0] <- 1

# Test whether samples within the same species are more similar than those not
mantel(iris.md ~ iris.model, nperm=10000)

## End(Not run)
```

---

mgram *Mantel correlogram*

Description

Calculates simple and partial Mantel correlograms.

Usage

```
mgram(species.d, space.d, breaks, nclass, stepsize, nperm = 1000,
    mrank = FALSE, nboot = 500, pboot = 0.9, cboot = 0.95,
    alternative = "two.sided", trace = FALSE)
```

Arguments

species.d	lower-triangular dissimilarity matrix.
space.d	lower-triangular matrix of geographic distances.
breaks	locations of class breaks. If specified, overrides nclass and stepsize.
nclass	number of distance classes. If not specified, Sturge's rule will be used to determine an appropriate number of classes.
stepsize	width of each distance class. If not specified, nclass and the range of space.d will be used to calculate an appropriate default.
nperm	number of permutations to use. If set to 0, the permutation test will be omitted.
mrank	if this is set to F (the default option), Pearson correlations will be used. If set to T, the Spearman correlation (correlation ranked distances) will be used.
nboot	number of iterations to use for the bootstrapped confidence limits. If set to 0, the bootstrapping will be omitted.

pboot	the level at which to resample the data for the bootstrapping procedure.
cboot	the level of the confidence limits to estimate.
alternative	default is "two.sided", and returns p-values for H0: rM = 0. The alternative is "one.sided", which returns p-values for H0: rM <= 0.
trace	if TRUE, returns progress indicators.

Details

This function calculates Mantel correlograms. The Mantel correlogram is essentially a multivariate autocorrelation function. The Mantel r represents the dissimilarity in variable composition (often species composition) at a particular lag distance.

Value

Returns an object of class mgram, which is a list with two elements. mgram is a matrix with one row for each distance class and 6 columns:

lag	midpoint of the distance class.
ngroup	number of distances in that class.
mantelr	Mantel r value.
pval	p-value for the test chosen.
llim	lower bound of confidence limit for mantelr.
ulim	upper bound of confidence limit for mantelr.

resids is NA for objects calculated by mgram().

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

References

Legendre, P. and M. Fortin. 1989. Spatial pattern and ecological analysis. Vegetatio 80:107-138.

See Also

mantel, plot.mgram, pmgram

Examples

```
## Not run:

# generate a simple surface
x <- matrix(1:10, nrow=10, ncol=10, byrow=FALSE)
y <- matrix(1:10, nrow=10, ncol=10, byrow=TRUE)
z <- x + 3*y
image(z)

# analyze the pattern of z across space
```

```
space <- cbind(as.vector(x), as.vector(y))
z <- as.vector(z)
space.d <- distance(space, "eucl")
z.d <- distance(z, "eucl")
z.mgram <- mgram(z.d, space.d, nperm=0)
plot(z.mgram)

## End(Not run)
```

MRM *Multiple Regression on distance Matrices*

Description

Multiple regression on distance matrices (MRM) using permutation tests of significance for regression coefficients and R-squared.

Usage

```
MRM(formula = formula(data), data = sys.parent(), nperm = 1000, mrank = FALSE)
```

Arguments

formula	formula in R/S-Plus format describing the test to be conducted.
data	an optional dataframe containing the variables in the model as columns of dissimilarities. By default the variables are taken from the current environment.
nperm	number of permutations to use. If set to 0, the permutation test will be omitted.
mrank	if this is set to FALSE (the default option), Pearson correlations will be used. If set to TRUE, the Spearman correlation (correlation ranked distances) will be used.

Details

Performs multiple regression on distance matrices following the methods outlined in Legendre et al. 1994.

Value

coef	A matrix with regression coefficients and associated p-values from the permutation test (using the pseudo-t of Legendre et al. 1994).
r.squared	Regression R-squared and associated p-value from the permutation test.
F.test	F-statistic and p-value for overall F-test for lack of fit.

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

References

Lichstein, J. 2007. Multiple regression on distance matrices: A multivariate spatial analysis tool. Plant Ecology 188: 117-131.

Legendre, P.; Lapointe, F. and Casgrain, P. 1994. Modeling brain evolution from behavior: A permutational regression approach. Evolution 48: 1487-1499.

See Also

mantel

Examples

```
data(graze)
LOAR10.mrm <- MRM(dist(LOAR10) ~ dist(sitelocation) + dist(forestpct), data=graze, nperm=100)
```

---

nmds *Non-metric multidimensional scaling*

Description

Non-metric multidimensional scaling.

Usage

```
nmds(dmat, mindim = 1, maxdim = 2, nits = 10, iconf = 0, epsilon = 1e-12,
    maxit = 500, trace = FALSE, stresscalc="default")
```

Arguments

dmat	lower-triangular dissimilarity matrix.
mindim	optional, minimum number of dimensions to use.
maxdim	optional, maximum number of dimensions to use.
nits	optional, number of separate ordinations to use.
iconf	optional, initial configuration. If not specified, then a random configuration is used.
epsilon	optional, acceptable difference in stress.
maxit	optional, maximum number of iterations.
trace	if TRUE, will write progress indicator to the screen.
stresscalc	if "default" uses modified stress calculation; if "kruskal" uses unmodified Kruskal stress calculation.

Details

The goal of NMDS is to find a configuration in a given number of dimensions which preserves rank-order dissimilarities as closely as possible. The number of dimensions must be specified in advance. Because NMDS is prone to finding local minima, several random starts must be used. Stress is used as the measure of goodness of fit. A lower stress indicates a better match between dissimilarity and ordination.

Value

`conf`	list of configurations.
`stress`	list of final stress values.
`r2`	total variance explained by each configuration.

The first results are for the lowest number of dimensions (total number is (mindim - maxdim + 1) * nits).

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

References

Kruskal, J.B. 1964. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. Psychometrika 29:1-27.

Minchin, P.R. 1987. An evaluation of the relative robustness of techniques for ecological ordination. Vegetatio 96:89-108.

See Also

`nmds.min`, `vf`

Examples

```
## Not run:
# Example of multivariate analysis using built-in iris dataset
data(iris)
iris <- iris[seq(1, 150, by=3),]
iris.md <- distance(iris[,1:4], "mahal")

# Minimum-stress 2-dimensional nonmetric multidimensional scaling configuration
# Uses small number of separate ordinations (5) to increase speed of example.
# Use more for final analysis.
iris.nmds <- nmds(iris.md, mindim=2, maxdim=2, nits=3)
iris.nmin <- nmds.min(iris.nmds)

# Plot NMDS result with symbols denoting species
plot(iris.nmin, pch=as.numeric(iris[,5]))

# Fit vectors for the main variables to the NMDS configuration
```

```
iris.vf <- vf(iris.nmin, iris[,1:4], nperm=10)
plot(iris.vf, col="blue")

## End(Not run)
```

---

nmds.min *Find minimum stress configuration*

Description

Finds minimum stress configuration from output of nmds()

Usage

```
nmds.min(x, dims = 0)
```

Arguments

x	output from nmds()
dims	desired dimensionality of result. If dims == 0 (the default) then the overall minimum stress configuration is chosen.

Value

Configuration of minimum stress ordination (dataframe of coordinates). The stress and r^2 for the minimum stress configuration are stored as attributes.

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

See Also

nmds

Examples

```
## Not run:
# Example of multivariate analysis using built-in iris dataset
data(iris)
iris <- iris[seq(1, 150, by=3),]
iris.md <- distance(iris[,1:4], "mahal")

# Minimum-stress 2-dimensional nonmetric multidimensional scaling configuration
# Uses small number of separate ordinations (5) to increase speed of example.
# Use more for final analysis.
iris.nmds <- nmds(iris.md, mindim=2, maxdim=2, nits=3)
iris.nmin <- nmds.min(iris.nmds)
```

```
# Plot NMDS result with symbols denoting species
plot(iris.nmin, pch=as.numeric(iris[,5]))

# Fit vectors for the main variables to the NMDS configuration
iris.vf <- vf(iris.nmin, iris[,1:4], nperm=10)
plot(iris.vf, col="blue")

## End(Not run)
```

---

pco *Principal coordinates analysis*

Description

Principal coordinates analysis (classical scaling).

Usage

```
pco(x, negvals = "zero", dround = 0)
```

Arguments

x	a lower-triangular dissimilarity matrix.
negvals	if = "zero" sets all negative eigenvalues to zero; if = "rm" corrects for negative eigenvalues using method 1 of Legendre and Anderson 1999.
dround	if greater than 0, attempts to correct for round-off error by rounding to that number of places.

Details

PCO (classical scaling, metric multidimensional scaling) is very similar to principal components analysis, but allows the use of any dissimilarity metric.

Value

values	eigenvalue for each component. This is a measure of the variance explained by each dimension.
vectors	eigenvectors. Each column contains the scores for that dimension.

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

See Also

princomp, nmds

Examples

```
## Not run:
data(iris)
iris.md <- distance(iris[,1:4], "mahal")
iris.pco <- pco(iris.md)

# scatterplot of the first two dimensions
plot(iris.pco$vectors[,1], iris.pco$vectors[,2], pch=as.numeric(iris[,5]))

## End(Not run)
```

`permresults` *Mantel test results for number of permutations, correlation and dataset size.*

Description

These results explore the variability in randomization tests of significance and in bootstrap confidence limits for different sample sizes, strengths of relationship and numbers of permutations.

Usage

```
data(permresults)
```

Format

A data frame with 15000 observations on the following 17 variables.

`n` The number of samples.

`r` The Mantel r of the dissimilarity vectors.

`p500` The p-values generated using 500 permutations.

`l100` The lower confidence limit generated using 100 subsamples.

`u100` The upper confidence limit generated using 100 subsamples.

`p1000` p-values with 1,000 permutations

`l500` lower CI with 500 permutations

`u500` upper CI with 500 permutations

`p10000` p-values with 10,000 permutations

`l1000` lower CI with 1,000 permutations

`u1000` upper CI with 1,000 permutations

`p100000` p-values with 100,000 permutations

`l10000` lower CI with 10,000 permutations

`u10000` upper CI with 10,000 permutations

`p1000000` p-values with 1,000,000 permutations

`l100000` lower CI with 100,000 permutations

`u100000` upper CI with 100,000 permutations

Details

These simulation results are included here because the entire simulation takes an extremely long time to run. The tests explored how the p-value and confidence limits varied with:

- number of samples (10 (45 dissimilarities), 25 (300), 50 (1225), 100 (4950))
- number of permutations: 500, 1000, 10000, 100000, 1000000
- number of bootstrap samples: 100, 500, 1000, 10000, 100000
- correlation of the original data: 0.01, 0.05, 0.10, 0.25, 0.50

The larger correlation values were omitted for the larger sample sizes because they were always highly signficant, and an additional low value of 0.025 was added for the largest sample size.

The method used to generate the data is shown in the example section.

Examples

```
data(permresults)

## Not run:
# Simulation Method
# example for sample size 10, correlation of 0.01

permresults <- data.frame(matrix(0, nrow=1000, ncol=17))
colnames(permresults) <- c("n", "r", "p500", "l100", "u100", "p1000",
    "l500", "u500", "p10000", "l1000", "u1000", "p100000", "l10000", "u10000",
    "p1000000", "l100000", "u100000")

# generate correlated data
corxy.len010.r010 <- corgen(len= 10, r= 0.01, epsilon=0.0001)

all.nperm <- c(500, 1000, 10000, 100000, 1000000)
all.nboot <- c(100, 500, 1000, 10000, 100000)

for(i in 1:1000) {
    permresults[i, 1] <- 10
    permresults[i, 2] <- 0.01
    for(j in 1:5) {
# record p-value, lower and upper confidence limits
        thismantel <- mantel(corxy.len010.r010$y ~ corxy.len010.r010$x,
  nperm=all.nperm[j], nboot=all.nboot[j])
        permresults[i, (3*j):(3*j + 2)] <- c(thismantel[c(2,5,6)])
    }
}

## End(Not run)
```

plot.mgram *Plot a Mantel correlogram*

Description

Plot a Mantel correlogram from an object of S3 class mgram, using solid symbols for significant values.

Usage

```
## S3 method for class 'mgram'
plot(x, pval = 0.05, xlab = "Distance", ylab = "Mantel r", ...)
```

Arguments

x	an object of class mgram
pval	cut-off level for statistical significance.
xlab	x-axis label.
ylab	y-axis label.
...	optional, any additional graphics parameters.

Value

draws a plot (graphics device must be active).

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

See Also

mgram

Examples

```
## Not run:
# generate a simple surface
x <- matrix(1:10, nrow=10, ncol=10, byrow=FALSE)
y <- matrix(1:10, nrow=10, ncol=10, byrow=TRUE)
z <- x + 3*y
image(z)

# analyze the pattern of z across space
space <- cbind(as.vector(x), as.vector(y))
z <- as.vector(z)
space.d <- distance(space, "eucl")
z.d <- distance(z, "eucl")
z.mgram <- mgram(z.d, space.d, nperm=0)
```

```
plot(z.mgram)

## End(Not run)
```

---

plot.vf *Plots fitted vectors onto an ordination diagram*

Description

Add vector fitting arrows to an existing ordination plot.

Usage

```
## S3 method for class 'vf'
plot(x, pval = 1, cex = 0.8, ascale = 0.9, ...)
```

Arguments

x	an object of S3 class vf, created by vf()
pval	optional, critical p-value for choosing variables to plot
cex	text size
ascale	optional, proportion of plot area to use when calculating arrow length
...	optional, other graphics parameters

Value

Adds arrows to an existing ordination plot. Only arrows with a p-value less than pval are added. By default, all variables are shown.

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

See Also

vf

Examples

```
## Not run:
# Example of multivariate analysis using built-in iris dataset
data(iris)
iris <- iris[seq(1, 150, by=3),]
iris.md <- distance(iris[,1:4], "mahal")

# Minimum-stress 2-dimensional nonmetric multidimensional scaling configuration
# Uses small number of separate ordinations (5) to increase speed of example.
```

```
# Use more for final analysis.
iris.nmds <- nmds(iris.md, mindim=2, maxdim=2, nits=3)
iris.nmin <- nmds.min(iris.nmds)

# Plot NMDS result with symbols denoting species
plot(iris.nmin, pch=as.numeric(iris[,5]))

# Fit vectors for the main variables to the NMDS configuration
iris.vf <- vf(iris.nmin, iris[,1:4], nperm=10)
plot(iris.vf, col="blue")

## End(Not run)
```

---

pmgram *Partial Mantel correlogram*

Description

This function calculates simple and partial multivariate correlograms.

Usage

```
pmgram(data, space, partial, breaks, nclass, stepsize, resids = FALSE, nperm = 1000)
```

Arguments

data	lower-triangular dissimilarity matrix. This can be either an object of class dist (treated as one column) or a matrix or data frame with one or two columns, each of which is an independent lower-triangular dissimilarity in vector form.
space	lower-triangular matrix of geographic distances.
partial	optional, lower-triangular dissimilarity matrix of ancillary data.
breaks	locations of class breaks. If specified, overrides nclass and stepsize.
nclass	number of distance classes. If not specified, Sturge's rule will be used to determine an appropriate number of classes.
stepsize	width of each distance class. If not specified, nclass and the range of space.d will be used to calculate an appropriate default.
resids	if resids=TRUE, will return the residuals for each distance class. Otherwise returns 0.
nperm	number of permutations to use. If set to 0, the permutation test will be omitted.

Details

This function does four different analyses: If data has 1 column and partial is missing, calculates a multivariate correlogram for data.

If data has 2 columns and partial is missing, calculates Mantel r between the two columns for each distance class.

If data has 1 column and partial exists, does a multivariate correlogram for the residuals, taking residuals over whole extent.

If data has 2 columns and partial exists, does a partial multivariate correlogram, calculating partial for each distance class separately.

Value

Returns a object of class mgram, which is a list containing two objects: mgram is a matrix with one row for each distance class and 4 columns:

`lag`	midpoint of the distance class.
`ngroup`	number of distances in that class.
`piecer`	Mantel r value.
`pval`	two-sided p-value.

resids is a vector of the residuals (if calcuated) and can be accessed with the residuals() method.

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

See Also

`mgram`, `mantel`

Examples

```
## Not run:
# generate a simple surface
x <- matrix(1:10, nrow=10, ncol=10, byrow=FALSE)
y <- matrix(1:10, nrow=10, ncol=10, byrow=TRUE)
z1 <- x + 3*y
z2 <- 2*x - y

# look at patterns
par(mfrow=c(1,2))
image(z1)
image(z2)

# analyze the pattern of z across space
z1 <- as.vector(z1)
z2 <- as.vector(z2)
z1.d <- distance(z1, "eucl")
```

```
z2.d <- distance(z2, "eucl")

space <- cbind(as.vector(x), as.vector(y))
space.d <- distance(space, "eucl")

# take partial correlogram of z2 on the residuals of z1 ~ space.d
z.pmgram <- pmgram(z1.d, space.d, z2.d, nperm=0)
par(mfrow=c(1,1))
plot(z.pmgram, pval=0.1)

## End(Not run)
```

`residuals.mgram` *Residuals of a Mantel correlogram*

Description

Extracts residuals from an S3 object of class `mgram` (only relevant for objects created by `pmgram{}`).

Usage

```
## S3 method for class 'mgram'
residuals(object, ...)
```

Arguments

`object`	an object of class `mgram`
`...`	additional arguments

Value

vector of residuals.

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

See Also

`mgram`

Examples

```
## Not run:
z.pmgram <- pmgram(z.d, space.d, resids=TRUE)
residuals(z.pmgram)

## End(Not run)

# A full example is available in the Mantel correlogram
# section of the main help file for \link{ecodist}.
```

---

vf *Vector fitting*

Description

Fits environmental variables to an ordination configuration.

Usage

```
vf(ord, vars, nperm = 100)
```

Arguments

ord	matrix containing a 2-dimensional ordination result with axes as columns.
vars	matrix with ancillary variables as columns.
nperm	number of permutation for the significance test. If nperm = 0, the test will be omitted.

Details

Vector fitting finds the maximum correlation of the individual variables with a configuration of samples in ordination space.

Value

an object of class vf containing matrix with the first 2 columns containing the scores for every variable in each of the 2 dimensions of the ordination space. r is the maximum correlation of the variable with the ordination space, and pval is the result of the permutation test.

Author(s)

Sarah Goslee, Sarah.Goslee@ars.usda.gov

References

Jongman, R.H.G., C.J.F. ter Braak and O.F.R. van Tongeren. 1995. Data analysis in community and landscape ecology. Cambridge University Press, New York.

See Also

plot.vf

Examples

```
## Not run:
# Example of multivariate analysis using built-in iris dataset
data(iris)
iris <- iris[seq(1, 150, by=3),]
iris.md <- distance(iris[,1:4], "mahal")

# Minimum-stress 2-dimensional nonmetric multidimensional scaling configuration
# Uses small number of separate ordinations (5) to increase speed of example.
# Use more for final analysis.
iris.nmds <- nmds(iris.md, mindim=2, maxdim=2, nits=3)
iris.nmin <- nmds.min(iris.nmds)

# Plot NMDS result with symbols denoting species
plot(iris.nmin, pch=as.numeric(iris[,5]))

# Fit vectors for the main variables to the NMDS configuration
iris.vf <- vf(iris.nmin, iris[,1:4], nperm=10)
plot(iris.vf, col="blue")

## End(Not run)
```

Index

∗Topic **datasets**
bump, 6
graze, 12
permresults, 24
∗Topic **distribution**
corgen, 7
∗Topic **hplot**
plot.mgram, 26
plot.vf, 27
∗Topic **manip**
crosstab, 8
fixdmat, 11
full, 11
lower, 15
∗Topic **multivariate**
bcdist, 5
cor2m, 6
distance, 10
ecodist-package, 2
mantel, 15
mgram, 17
MRM, 19
nmds, 20
nmds.min, 22
pco, 23
plot.mgram, 26
plot.vf, 27
pmgram, 28
residuals.mgram, 30
vf, 31

bcdist, 5, *11*
bump, 6

cor2m, 6
corgen, 7
crosstab, 8

dist, *5*, *10, 11*
distance, 10

ecodist *(*ecodist-package*)*, 2
ecodist-package, 2

fixdmat, 11, *15*
full, *11*, 11, *15*

graze, 12

lower, *11, 12*, 15

mantel, 15, *18*, *20*, *29*
mgram, *17*, 17, *26*, *29, 30*
MRM, 19

nmds, 20, *22, 23*
nmds.min, *21*, 22

pco, 23
permresults, 24
plot.mgram, *18*, 26
plot.vf, 27, *32*
pmgram, *18*, 28
princomp, *23*

residuals.mgram, 30

vf, *21*, *27*, 31