Package 'Biolinv'

January 3, 2018

Type Package

Title Modelling and Forecasting Biological Invasions

Version 0.1-2

Author Luca Butikofer [aut, cre],
Beatrix Jones [aut]

Maintainer Luca Butikofer <lucabutikofer@gmail.com>

Description Analysing and forecasting biological invasions time series with a stochastic, non-mechanistic approach that gives proper weight to the anthropic component, accounts for habitat suitability and provides measures of precision for its estimates.

License GPL-3

LazyData TRUE

Depends R (>= 3.2.4)

Imports raster (>= 2.5-2), fields (>= 8.3-6), spatstat (>= 1.48-0), sp (>= 1.2-4), grDevices (>= 3.3.2), stats (>= 3.3.2), classInt (>= 0.1-23)

RoxygenNote 5.0.1

Date 2017-02-02

NeedsCompilation no

Repository CRAN

Date/Publication 2018-01-03 15:21:35 UTC

R topics documented:

`cosd` *Computes the cosine of an angle expressed in decimal degrees.*

Description

Computes the cosine of an angle expressed in decimal degrees.

Usage

```
cosd(degrees)
```

Arguments

`degrees`	either a number or a numeric vector of angles in decimal degrees.

Value

number or numeric vector.

Author(s)

Luca Butikofer

Examples

```
cosd(90)
plot(seq(0,360,1), cosd(seq(0,360,1)), type='l')
```

EM *Runs the EM algorithm.*

Description

Attributes to the populations in an invasion time series a probability value of being of natural origin, as opposite of anthropogenic origin.

Usage

```
EM(dataset, randompoints, sigma, pi)
```

Arguments

dataset	the data frame to be analised (WGS84, colums order should be "year","lat","long","origin")
randompoints	data frame of 'y' and 'x' coordinates of random points (projected coordinate system)
sigma	starting value for the standard deviation of the natural dispersal kernel (assumed to be a half normal)
pi	starting value for the proportion of natural points in the dataset

Value

dataset argument with two additional colums. 'dist': distance from nearest point of natural origin or nearest anchor point (see Details); 'Pnat': probability of being of natural origin.

Author(s)

Beatrix Jones, Luca Butikofer

Examples

```
data('nzp')
data('frogs')
randp<- RPG(rpopn=1000, boundary=nzp, SP= 'random_frog')
frogsEM<- EM(dataset= frogs, randompoints= randp, sigma=6, pi=0.5)
```

frogs *Frogs sighting locations.*

Description

Dataset containing sighting locations of L. raniformis in New Zealand.

Usage

```
data(frogs)
```

Format

A dataframe with 194 rows and 4 columns

Details

- year: year of the sighting
- y: latitude in the New Zealand Transverse Mercatore coordinate system (length unit of measure: meter).
- x: longitude in the New Zealand Transverse Mercatore coordinate system (length unit of measure: meter).
- species: the name of the species.

Source

Selection of sightings locations form the Herpetofauna Database of New Zealand. Kindly provided by Benno Kappers, Department of Conservation.

frogsEM *Frogs sighting locations and the output of the EM() function.*

Description

Version of 'frogs' Dataset containing sighting locations of L. raniformis in New Zealand as returned by function EM().

Usage

```
data(frogsEM)
```

Format

A dataframe with 194 rows and 6 variables

Details

- year: year of the sighting
- y: latitude in the New Zealand Transverse Mercatore coordinate system (length unit of measure: meter).
- x: longitude in the New Zealand Transverse Mercatore coordinate system (length unit of measure: meter).
- species: the name of the species.
- Pnat: probability of being of natural origin [0;1].
- Dist: distance from nearest point of natural origin or form anchor point.

---

frogsJK — *Four Jackknife re-samplings done on dataset 'frogs' generated by function jackKnife().*

Description

List of four Jackknife re-samplings of dataframe 'frogs' as returned by function jackKnife()(see examples in ?jackKnife).

Usage

```
data(frogsJK)
```

Format

A list of four dataframes with 164 rows and 6 columns each.

Details

- year: year of the sighting
- y: latitude in the New Zealand Transverse Mercatore coordinate system (length unit of measure: meter).
- x: longitude in the New Zealand Transverse Mercatore coordinate system (length unit of measure: meter).
- species: the name of the species.

`frogsLacro` *Simulated datasets for comparison with 'frogs' dataset generated with function simulacro() (see examples in ?simulacro).*

Description

List of eight lists (one per different Alpha value used) each containing ten dataframes of simulated sigthing locations built to simulate 'frogs' dataset.

Usage

```
data(frogsLacro)
```

Format

list of 8 lists of 10 dataframes each with 164 rows and 6 columns each.

Details

- year: year of the sighting
- y: latitude in the New Zealand Transverse Mercatore coordinate system (length unit of measure: meter).
- x: longitude in the New Zealand Transverse Mercatore coordinate system (length unit of measure: meter).
- species: the name of the species.
- Pnat: probability of being of natural origin [0;1].
- Dist: distance from nearest point of natural origin or form anchor point.

`frogsSum` *Summary of the dissimilarity values of each of the dataframes in 'frogsLacro' with dataframe 'frogs'.*

Description

Dataframe with dissimilarity values and respective Alpha value as obtained by function modSel() (see examples in ?modSel).

Usage

```
data(frogsSum)
```

Format

list of 8 lists of 10 dataframes each with 164 rows and 6 columns each.

Details

- dissimilarity: dissimilarity value
- compAlpha: alpha value of the comparison dataset.

`fx` *One dimentional dispersal kernel function.*

Description

Computes user-defined one dimensional dispersal kernels.

Usage

```
fx(x, a, c)
```

Arguments

`x`	vector of distances.
`a`	Alpha value.
`c`	C value.

Value

numeric vector.

Author(s)

Luca Butikofer

Examples

```
plot(1:1000, fx(x= 1:1000, a= 300, c=2), type='l',
 ylab= 'Probablitiy', xlab='Distance',
 main= 'One Dimensional Dispersal Kernel \n Alpha= 300; C= 2')
```

jackKnife *Jackknifes a dataset.*

Description

This function performes jackknife resampling on a dataset.

Usage

```
jackKnife(DF, N, PR = 0.85, DIR = F)
```

Arguments

DF	data frame, matrix or any R object wich responds to function rownames().
N	number of desired jackknifed datasets.
PR	proportion of entries to DF that will be kept in the jackknifed datasets. Default is 0.85.
DIR	directory whre to save the jackknifed datasets. If FALSE (default) will not save to disk.

Value

list of jacknifed datasets. If DIR is specified also a folder in directory DIR containing one .RDS file per jackknifed dataset (with extension .jds - Jackknifed Data Set) will be created.

Author(s)

Luca Butikofer

Examples

```
data('frogs')
frogsJK<- jackKnife(DF= frogs, N= 10)
```

modSel *Wrapper for function spatSim() which allows use on multiple datasets.*

Description

This function uses spatSim() function to compare many simulated datasets with one, typically observed, data set.

Usage

```
modSel(WIN, M0, M2, AV, RAD)
```

Arguments

WIN	window of observation for the point patterns (MOD0 and MOD2)(see ?spatstat::owin). Object of class 'sp::owin'.
M0	data frame containing 'y' and 'x' coordinates (projected coordinate system) to compare with M2.
M2	list of dataframes with same structure of M0, typically generated with simulacro() function.
AV	numeric vector of the Alpha values of the simulated datasets in the same order as in the list of argument M2. Used to save the output data frame.
RAD	numeric vector of search distances for the K-function.

Value

data frame with two columns. 'dissimilarity': dissimilarity values as computed by spatSim() function. 'compAlpha': same as AV.

Author(s)

Luca Butikofer

Examples

```
data(nzw)
data(frogsEM)  #see EM().
data(frogsLacro)  #see simulacro().

## Not run:
frogsSum<- modSel(WIN= nzw, M0= frogsEM, M2= frogsLacro,
 AV= c(2,3,4.5,7.5,11,15,20,25), RAD= seq(0,30000,1000))

## End(Not run)
```

NPG *Generates points of natural origin.*

Description

This function generates points of natural origin around existent points by sampling their distance from the original propagule from a user-defined dispersal kernel (as set in function SDS(), the output of which is to be entered in this function as argument 'Delta'.

Usage

```
NPG(PresLoc, N, Deltas, year)
```

Arguments

PresLoc	data frame with coordinates of sighting locations. Column 2 must be 'y' (latitude on a projected coordinate system with meters as distance unit of measure) and column 3 must be 'x' (longitude on a projected coordinate system with meters as distance unit of measure).
N	total number of points to generate.
Deltas	y and x shifts as generated from function SDS(). This function samples distance and angle of new points form existing ones from this argument.
year	the year lable to attach to the generated points

Value

data frame of combined source ('PresLoc' argument) and new locations.

Author(s)

Luca Butikofer

---

nzp *Landmasses of the two main islands of New Zealand as SpatialPolygons.*

Description

Object of class sp::SpatialPolygons. North and South Islands of New Zealand in the New Zealand Transverse Mercatore projection.

Usage

```
data(nzp)
```

Format

object of class sp::SpatialPolygons.

Source

mapdata::worldHires()

nzw — *Landmasses of the two main islands of New Zealand as Window object.*

Description

Object of class spatstat::owin of the North and South Islands of New Zealand in the New Zealand Transverse Mercatore projection obtained from object 'nzp' (see ?nzp).

Usage

```
data(nzw)
```

Format

object of class spatstat::owin.

plotAlpha — *Plots the output of modSel().*

Description

This function makes a plot of the data frame outputted by function modSel(). Useful to choose wich of the simulated datasets is the most similar to the observed one.

Usage

```
plotAlpha(SSIM, REP, BP = FALSE)
```

Arguments

SSIM	data frame output of modSel.
REP	number of replicates.
BP	when FALSE (default) plots a line representing the average of all replicates. When TRUE plots boxplots of all replicates for each alpha value.

Value

plot.

Author(s)

Luca Butikofer

Examples

```
data(frogsSum)
plotAlpha(SSIM= frogsSum, REP= 10)
plotAlpha(SSIM= frogsSum, REP= 10, BP=TRUE)
```

plotlacro *Plots the file type used as input and output of simulacro().*

Description

This function makes a plot of the data frame used for argument INIDIST of function simulacro() and of the output of simulacro(). The plot shows the points as circles and crosses to represent their natural and anthropic origin respectively and has an iformative legend.

Usage

```
plotlacro(x, tr = 0.5, outline, main = "std")
```

Arguments

x	simulacro-type file (as outputted by simulacro() or inputted as argument INIDIST).
tr	threshold for Pnat above which points are considered of human nature.
outline	geographical boundaries to x. Object of class sp::SpatialPolygons or data frame or matrix with the outline's corners.
main	main title of the plot, by default is 'Map of ...' where ...= x$species (the name of the modelled species).

Value

plot.

Author(s)

Luca Butikofer

Examples

```
data(frogsEM)
plotlacro(x= frogsEM, outline= nzp)
```

RPG *Generates random points within a geographic boundary.*

Description

This function generates random points within a geographic area set as either sp::SpatialPolygons or raster::RasterLayer.

Usage

```
RPG(rpopn, boundary, year = "anyYear", SP)
```

Arguments

rpopn	total number of random points to be generated.
boundary	either a RasterLayer object (0: inside; >= 1: outside) or a SpatialPolygon(s) object.
year	year name (used by wrapper function simulacro() to lable consecutive time steps in its simulate time series).
SP	name of the species being simulated for labelling the output.

Value

data frame of random points.

Author(s)

Luca Butikofer

SDS *Generates a list of dispersal vectors.*

Description

This function generates a set of spatial vectors whose length is sampled form a user-defined dispersal kernel and whose direction is sampled randomly from 1 degree to 360 degrees.

Usage

```
SDS(probDist, DIST, npop = 10000)
```

Arguments

probDist	Probability Distribution of npop.
DIST	vector of possible distances [m] to be sampled for dispersal.
npop	number of distance-angle couples that need to be produced.

Value

data frame of y and x shifts.

Author(s)

Luca Butikofer

Examples

```
dist<- seq(.1, 30, .1)
prob<- fx(x=dist, a=7, c=2)
deltas<- SDS(DIST=dist, probDist=prob)
```

---

simulacro *Builds a simulated biological invasion dataset.*

Description

This function builds point time seris within geographic borders based on an iterative process that simulates biological invasions. Points of natural and anthropic origin can be generated with different processes. Points of natural origin are generated by a stepwise process where new locatins are chosen on the basis of a user-definable dispersal kernel. Points of antrhopic origin are sampled randomly. All generated points can be filtered through probability maps.

Usage

```
simulacro(INIDIST, YEARS, BOUNDARY, NNAT, NANTH, A, C = 2, X, HSM = FALSE,
  FACANTH = 1, FACNAT = 1, ITERATIONS = 1, DIR = F,
  TRUEANTH = c(FALSE, TRUE), TRUEDB, PROB = 0.5)
```

Arguments

INIDIST	data frame of initial distribution. Columns must be: 'year' (year of first sighting); 'y' (latitude on a projected coordinate system with meters as distance unit of measure); 'x' (longitude on a projected coordinate system with meters as distance unit of measure); 'species' (the name of the species for that sighting location); 'Pnat' ([0;1], probability for that sighting location of being of natural origin (as computed by function EM())), 'Dist' (dispersal distance of that sighting location as computed by function EM()).
YEARS	vector of unique year values for which the simulacro function will generate data (they will be written in the output file).
BOUNDARY	object of class sp::SpatialPolygons or raster:RasterLayer (0: inside, >= 1: outside) used as geographi boundary for function RPG(). These are the geographic limits within which points or anthropic origin can be generated.
NNAT	either a vector of numbers of natural points to be generated every year or an integer number of points to be generated per time-step.

NANTH	either a vector of numbers of anthropic points to be generated every year or an integer number of points to be generated per time-step.
A	Alpha values for the one dimensional dispersal kernel. Each Alpha value will be used to generate a single data frame.
C	C values for the one dimensional dispersal kernel. C=2: normal kernel; C=1: negative exponenetial kernel; C<1: fat-tailed kernel.
X	set of possible interpopulation distances [km].
HSM	either a Habitat Suitability Map ([0;1], object of class raster::RasterLayer, probability raster giving value of likelyhood of a viable population establishing each cell) or the geographic boundaries (object of class sp::SpatialPolygons) within which to generate points of natural or anthropic origin.
FACANTH	Only if HSM is a probability raster. Factor multiplying NANTH before the filtering. This value should be high if HSM has a low proportion of suitable cells.
FACNAT	Only if HSM is a probability raster. Factor multiplying NNAT before the filtering
ITERATIONS	number of replicate datasets generated with the same Alpha and C values.
DIR	directory where to write the simulated datasets. if FALSE they will be saved as data frames in a list object.
TRUEANTH	If FALSE will use HSM to simulate anthropogenic dispersal. If TRUE will use TRUEDB
TRUEDB	data frame containing points of anthropic origin (must be in the same format as INIDIST, rows where TRUEDB$Pnat>PROB will be ignored). Ignored if TRUEANTH= FALSE, but HSM must then be specified.
PROB	threshold over which Pnat is considered natural.

Value

list of data frames. Every data frame represents a simulated biological invasion. if DIR=TRUE one folder will be created for each Alpha and C combination containig all the replicates datasets set in ITERATION. if DIR=FALSE (default) the order in the list will follow the order of the C and Alpha values respectively as set in C and A.

Author(s)

Luca Butikofer

Examples

```
data('frogsEM')  #see example in ?EM().
data('nzp')

idst<- frogsEM[1:10,]
Cr<- frogsEM[-(1:10),]
yr<- unique(Cr$year)
```

```
nNoYear<- rep(NA,length(unique(Cr$year)))
hNoYear<- rep(NA,length(unique(Cr$year)))

for(i in 1:length(unique(Cr$year))){
 CrYear<- Cr[Cr$year==unique(Cr$year)[i],]  #Cr for that year
 nNoYear[i]<- nrow(CrYear[CrYear$Pnat>=.5,])  #natural points for that year
 hNoYear[i]<- nrow(CrYear[CrYear$Pnat<.5,])  #human points for that year
}

AV<- c(2,3,4.5,7.5,11,15,20,25)  #alpha values

## Not run:
frogsLacro<- simulacro(INIDIST=idst,YEARS=yr,
 BOUNDARY=nzp,NNAT=nNoYear,NANTH=hNoYear,
 FACNAT=10,
 A=AV,X=seq(.1,30,.1),
 TRUEANTH=TRUE,TRUEDB=Cr,PROB=.5,
 ITERATIONS=10,HSM=nzp)

## End(Not run)
```

`sind` *Computes the sine of an angle expressed in decimal degrees.*

Description

Computes the sine of an angle expressed in decimal degrees.

Usage

```
sind(degrees)
```

Arguments

`degrees`	either a number or a numeric vector of angles in decimal degrees.

Value

number or numeric vector.

Author(s)

Luca Butikofer

Examples

```
sind(90)
plot(seq(0,360,1), sind(seq(0,360,1)),type='l')
```

`spatFilter` *Filters a set of points based on a probability map.*

Description

This function rarefies a set of points using a raster map of probabilities of a point persisting in that location.

Usage

```
spatFilter(points, MAP, Nclass = FALSE)
```

Arguments

`points`	data frame containing columns 'y' and 'x' (spatial coordinates on projected coordinate system).
`MAP`	object of class RasterLayer. Map of probability [0;1] of a point to persist in any pixel.
`Nclass`	optional number of classes into which MAP is going to be binned (quantile-based). This can be useful if the probability distribution of MAP is highly skewed.

Value

data frame with same structure as 'points' but with less rows (points).

Author(s)

Luca Butikofer

`spatSim` *Computes spatial dissimilarity of two point processes.*

Description

This function uses Rippley's K-function (see Details) to compute spatial dissimilarity of two point processes.

Usage

```
spatSim(MOD0, MOD2, WINDOW, R)
```

Arguments

MOD0	data frame containing 'y' and 'x' coordinates (projected coordinate system) to compare with MOD2.
MOD2	data frame containing 'y' and 'x' coordinates (projected coordinate system) to compare with MOD0.
WINDOW	window of observation of the point patterns (MOD0 and MOD2)(see ?spatstat::owin).Must be an object of class 'owin'.
R	numeric vector of searc distances for the K-function.

Value

squared sum of distances between K-functions. This is a measure of spatial dissimilarity.

Author(s)

Luca Butikofer

Examples

```
ran<- data.frame('y'=sample(1000),'x'=sample(1000))
nor<- data.frame('y'=rnorm(1000,sd=150,mean=500),'x'=rnorm(1000,sd=150,mean=500))
window<-  spatstat::owin(xrange=c(0,1000),yrange=c(0,1000))
spatSim(ran, nor, WINDOW= window, R=0:200)
```

Index