

Package ‘iSDM’

April 13, 2016

Type Package

Title Invasive Species Distribution Modelling

Version 1.0

Date 2016-04-05

Author

Hattab Tarek <hattab.tarek@gmail.com> & Jonathan Lenoir <jonathan.lenoir@u-picardie.fr>

Maintainer Hattab Tarek <hattab.tarek@gmail.com>

Description iSDM is an open-source R package that implements a few functions useful for modelling the spatial distribution of invasive species

Depends R (>= 3.2.0)

Suggests MASS

Imports ade4,
colorRamps,
geometry,
geosphere,
maptools,
MASS,
pdist,
raster,
rgdal,
rgl,
sp,
virtualspecies

License GPL-3

RoxygenNote 5.0.1

Vignettes SDM, Species distribution modelling, invasive species,RGB plot

R topics documented:

iSDM-package	2
EnvSysSample	2
invasionForce	4

likelihoodCA	5
RGBplot	7

Index	9
--------------	----------

iSDM-package	<i>Invasive Species Distribution Modelling</i>
--------------	--

Description

iSDM is an open-source R package that implements a few functions useful for modeling the spatial distribution of invasive species

Author(s)

Hattab Tarek <hattab.tarek@gmail.com> & Jonathan Lenoir <jonathan.lenoir@u-picardie.fr>

Maintainer: Hattab Tarek <hattab.tarek@gmail.com>

References

Tarek Hattab, Ruben Van De Kerchove, Ben Somers, Boris Brasseur, Carol Ximena Garzon Lopez, Duccio Rocchini, Emilie Gallet-Moron, Fabien Spicher, Guillaume Decocq, Hannes Feilhauer, Helene Horen, Jens Warrie, Michael Ewald, Olivier Honnay, Pieter Kempeneers, Raf Aerts, Sandra Skowronek, Sebastian Schmidlein and Jonathan Lenoir (In prep). An unified framework to model the potential and realized distributions of invasive species within the invaded range

EnvSysSample	<i>R function to perform a environmental systemic sampling design</i>
--------------	---

Description

R function to perform a environmental systemic sampling design

Usage

```
EnvSysSample(envData,nExpect,plot=TRUE,saveShape=TRUE,nf,lowerLim,upperLim)
```

Arguments

envData	May be a SpatialPointsDataFrame, a SpatilPixelsDataFrame, or a SpatialGrid-DataFrame as defined in package sp or a RasterStack or RasterBrick as defined in package raster. Note that this object can contain a mixed type variables (quantitative, factor and ordered)
nExpect	Numeric Number of desired sampling points. Note that the exact expected number can not be reached because this function tries to find a grid that best matches the expected number

plot	Logic TRUE if you would like to have a graphical representation of the environmental systemic sampling design
saveShape	Logic TRUE if you want to save the geographical coordinates of the pixels corresponding to this systemic sampling design as a shapefile format. The generated shapefile will be saved in your working directory
nf	Numeric The number of ordination axes retained in the analysis, this function manages currently only 2 or 3 dimensions
lowerLim	Numeric value of probability in [0,1] that can be used to produce sample quantiles corresponding to the given lower limit probability. This may be used to reduce the effect of extremes observations
upperLim	Numeric value of probability in [0,1] that can be used to produce sample quantiles corresponding to the given upper limit probability. This may be used to reduce the effect of extremes observations

Details

The environmental systemic sampling design consists in reducing the environmental space in 2 or 3 dimensions using an ordination method (the dudi.mixt method in ade4 package) as a first step. Thereafter convex hull will be created around the reduced environmental space. Then, a multidimensional grid will be created inside the convex hull. The obtained grid represents the perfect configuration required to adequately survey the environmental space. The last step is subsequently to seek the closest pixel to this ideal configuration. This is achieved by searching the nearest neighbour between each grid point and each pixel in the environmental space based on Euclidean distances

Value

Returns a list containing 4 objects: GeoSamples (The geographical coordinates of the pixels corresponding to the systemic sampling design); EnvSamples (The coordinates in the reduced environmental space of pixels corresponding to the systemic sampling design); EnvGrid (The coordinates of the grid in the reduced environmental); cellsize (the sizes of the grid cells in the different dimensions)

References

Tarek Hattab, Ruben Van De Kerchove, Ben Somers, Boris Brasseur, Carol Ximena Garz?n L?pez, Duccio Rocchini, Emilie Gallet-Moron, Fabien Spicher, Guillaume Decocq, Hannes Feilhauer, H?re Horen, Jens Warrie, Michael Ewald, Olivier Honnay, Pieter Kempeneers, Raf Aerts, Sandra Skowronek, Sebastian Schmidtlein and Jonathan Lenoir (In prep). An unified framework to model the potential and realized distributions of invasive species within the invaded range

Examples

```
library(raster)
envData<-getData('worldclim', var='bio', res=10)
envData<-crop(envData,extent(-10,45,20,75))
plot(envData)
par(mfrow=c(1,2))
Mysampling1<-EnvSysSample(envData,nExpect=50,plot=TRUE,saveShape=TRUE,nf=2,lowerLim=0.00001,upperLim=0.99999)
plot(envData[[1]])
```

```

plot(Mysampling1[[1]],add=TRUE,col=2,pch=19)
par(mfrow=c(1,2))
Mysampling2<-EnvSysSample(envData,nExpect=50,plot=TRUE,saveShape=TRUE,nf=2,lowerLim=0.1,upperLim=0.9)
plot(envData[[1]])
plot(Mysampling2[[1]],add=TRUE,col=2,pch=19)
Mysampling3<-EnvSysSample(envData,nExpect=50,plot=TRUE,saveShape=TRUE,nf=3,lowerLim=0.001,upperLim=0.999)
plot(envData[[1]])
plot(Mysampling3[[1]],add=TRUE,col=2)

```

invasionForce

R function to compute a negative exponential dispersal kernel

Description

R function to compute a negative exponential dispersal kernel

Usage

```
invasionForce(occData,coords,a,envData,binary=TRUE,longlat=NULL)
```

Arguments

occData	May be a SpatialPointsDataFrame as defined in package sp, a data.frame or a matrix object containing species data
coords	Optional 2 columns matrix containing the X and Y coordinates of occData(only consider if occData is a data.frame or a matrix object)
a	An integer between 0 and 1 that controls the form of the dispersal kernel
envData	May be a SpatialPointsDataFrame or a SpatialGridDataFrame as defined in package sp or a RasterStack or RasterBrick as defined in package raster. This object will be used to determine the area for which the negative exponential dispersal kernel will be calculated
binary	logical TRUE if the occData object contains presences/absences data and FALSE if occData contains only the geographical coordinates of presences data
longlat	Logical If TRUE, coordinates should be in degrees; else they should represent planar (Euclidean) space (e.g. units of meters)

Value

Returns a RasterLayer object of the negative exponential dispersal kernel

Examples

```

library(raster)
library(sp)
envData<-getData('worldclim', var='bio', res=10)
envData<-crop(envData,extent(-8,15,38,55))
envData<-envData[[1]]

```

```
#Generate randomly a SpatialPointsDataFrame containing occurrences
xy<-coordinates(envData)[sample(which(is.na(values(envData)))==FALSE),100),]
occ<-ifelse(xy[,2]>50,0,1)
occData<-SpatialPointsDataFrame(coords=xy,data=as.data.frame(occ), proj4string = CRS(proj4string(envData)))

par(mfrow=c(3,3),mar=c(1.5,1.5,1.5,1.5))
for (a in c(0.01,0.03,0.05,0.06,0.09,0.1,0.3,0.6,0.9)){
  propagule<-invasionForce(occData,envData=envData,a=a,binary=TRUE,longlat=TRUE)
  plot(propagule,main=paste("a = ",a))
  plot(occData,col=ifelse(occData@data[,1]==1,1,0),add=TRUE,cex=0.3)}
```

likelihoodCA	<i>R function to compute the likelihood of detecting contingent absences within a network of presence-absence data</i>
--------------	--

Description

R function to compute the likelihood of detecting contingent absences

Usage

```
likelihoodCA(occData,coords=NULL,envData, longlat=TRUE,nf=5)
```

Arguments

occData	May be a SpatialPointsDataFrame as defined in package sp, a data.frame or a matrix object containing species data (a single species) in binary format (ones for presences, zeros for absences)
coords	Optional 2 columns matrix containing the X and Y coordinates of occData (only consider if occData is a a data.frame or a matrix object)
envData	An abject containing your explanatory variables. May be a SpatialPointsDataFrame or SpatialGridDataFrame as defined in package sp or a RasterStack or RasterBrick as defined in package raster. Note that this object can contain a mixed type variables (quantitative, factor and ordered).
longlat	Logical TRUE if point coordinates are not projected(longitude-latitude decimal degrees)(+proj=longlat) and FALSE for plane (lonlat=FALSE)
nf	If envData contains any factor and ordered variables, the Hill and Smith method will be used to perform an ordination of explanatory variables. nf corresponds in this case to the number of kept axes for the calulation of mahalanobis distances.

Details

This function campute the likelihood of detecting contingent absences within a network of presence-absence data. It's based on the idea that absences data that are environmentally close but geographically distant to presences data are more likely to be contingent absences.

Value

Returns a `SpatialPointsDataFrame` containing the likelihood values

References

Tarek Hattab, Ruben Van De Kerchove, Ben Somers, Boris Brasseur, Carol Ximena Garz?n L?pez, Duccio Rocchini, Emilie Gallet-Moron, Fabien Spicher, Guillaume Decocq, Hannes Feilhauer, H?ne Horen, Jens Warrie, Michael Ewald, Olivier Honnay, Pieter Kempeneers, Raf Aerts, Sandra Skowronek, Sebastian Schmidtlein and Jonathan Lenoir (In prep). An unified framework to model the potential and realized distributions of invasive species within the invaded range

Examples

```
library(raster)
library(sp)
library(virtualspecies)
library(colorRamps)

envData<-getData('worldclim', var='bio', res=10)
envData<-crop(envData,extent(-8,15,38,55))

#Generate virtual species responses with formatfunctions
my.parameters <- formatFunctions(bio1 = c(fun = "dnorm", mean = 140, sd = 40), bio5 = c(fun = "dnorm",
mean = 230, sd = 70),bio6 = c(fun = "dnorm",mean = 10, sd = 40))

#Generate a virtual species distributions with responses to environmental variables
potential.dist <- generateSpFromFun(envData[[c(1,5,6)]], my.parameters)

#Limit a virtual species distribution to a defined area.
#It will thus generate a species which is not at the equilibrium with its environment
realized.dist<-limitDistribution(x=potential.dist$suitab.raster, area=extent(-8,15,38,48))

#Generate a random presence absence dataset from the realized distribution
# using a probability threshold of 0.5
Presence<-coordinates(realized.dist$occupied.area
)[sample(which(values(realized.dist$occupied.area)>0.5),300),]
Absence<-coordinates(realized.dist$occupied.area
)[sample(which(values(realized.dist$occupied.area)<0.5),300),]
occData<-as.data.frame(rbind(cbind(Presence,SP=rep(1,300)),cbind(Absence,SP=rep(0,300))))
coordinates(occData)<-~x+y
proj4string(occData)<-proj4string(envData)

# Calculat the likelihood of detecting contingent absences
likelihood<-likelihoodCA(occData=occData,envData=envData[[c(1,5,6)]],longlat=TRUE)

# Display the results
par(mfrow=c(1,2),mar=c(2,2.5,2,2.5))
plot(realized.dist$occupied.area,main="Realized distribution")
plot(occData,col=ifelse(occData$SP==1,2,1),add=TRUE,pch=19,cex=0.8)
plot(potential.dist$suitab.raster,main="Potential distribution")

scatterCol<-function(x){
```

```

x<-(x-min(x))/(max(x)-min(x))
colorFunction <- colorRamp(matlab.like(100))
zMatrix <- colorFunction(x)
zColors <- rgb(zMatrix[,1], zMatrix[,2], zMatrix[,3], maxColorValue=255)
return(zColors)}

points(likelihood,pch=21, col=1,bg=scatterCol(likelihood@data[, "LCA"]),cex=1)

```

RGBplot	<i>R function to plot the realized versus the potential distribution using a RGB colour space</i>
---------	---

Description

R to plot the realized versus the potential distribution using a RGB colour space

Usage

```
RGBplot(x=x,y=y,cex,xlab=xlab,ylab=ylab)
```

Arguments

x	May be a SpatilPixelsDataFrame, or a SpatialGridDataFrame as defined in package sp or a RasterLayer as defined in package raster
y	An object belonging to the same class as x
cex	A numerical value giving the amount by which plotting symbols should be magnified relative to the default
xlab	A title for the x axis
ylab	A title for the y axis

References

Tarek Hattab, Ruben Van De Kerchove, Ben Somers, Boris Brasseur, Carol Ximena Garz?n L?pez, Duccio Rocchini, Emilie Gallet-Moron, Fabien Spicher, Guillaume Decocq, Hannes Feilhauer, H<e9>l<e8>ne Horen, Jens Warrie, Michael Ewald, Olivier Honnay, Pieter Kempeneers, Raf Aerts, Sandra Skowronek, Sebastian Schmidtlein and Jonathan Lenoir (In prep).An unified framework to model the potential and realized distributions of invasive species within the invaded range

Examples

```

library(raster)
library(sp)
library(virtualspecies)
library(colorRamps)

envData<-getData('worldclim', var='bio', res=10)
envData<-crop(envData,extent(-8,15,38,55))

```

```
# Generate virtual species responses with formatfunctions
my.parameters <- formatFunctions(bio1 = c(fun = "dnorm", mean = 140, sd = 40), bio5 = c(fun = "dnorm",
mean = 230, sd = 70), bio6 = c(fun = "dnorm", mean = 10, sd = 40))
# Generate a virtual species potential distributions with responses to environmental variables
potential.dist <- generateSpFromFun(envData[[c(1,5,6)]], my.parameters)$suitab.raster

#Limit the distribution
realized.dist<-potential.dist
cell.id<-which(coordinates(realized.dist)[,2]>48)
dis.lim<-sample(seq(0,1,by=0.01),length(cell.id),replace=TRUE)
values(realized.dist)[cell.id]<-ifelse(values(realized.dist)[cell.id]>dis.lim,
values(realized.dist)[cell.id]-dis.lim,0)
RGBplot(x=potential.dist,y=realized.dist,cex=0.6,xlab="Potential distribution"
,ylab="Realized distribution")
```


Index

*Topic **SDM, Species distribution
modelling, invasive
species,RGB plot**
iSDM-package, [2](#)

EnvSysSample, [2](#)

invasionForce, [4](#)

iSDM (iSDM-package), [2](#)

iSDM-package, [2](#)

likelihoodCA, [5](#)

RGBplot, [7](#)