

Package ‘AHMbook’

May 19, 2017

Type Package

Title Functions and Data for the Book 'Applied Hierarchical Modeling in Ecology'

Version 0.1.4

Date 2017-05-19

Depends R (>= 2.10), unmarked (>= 0.12.2)

Imports grDevices, graphics, methods, stats, utils, plotrix, raster, sp, RandomFields, coda

Suggests

Author Marc Kery, Andy Royle, Mike Meredith

Maintainer Mike Meredith <mmeredith@wcs.org>

Description Provides functions and data sets to accompany the book 'Applied Hierarchical Modeling in Ecology: Analysis of distribution, abundance and species richness in R and BUGS' by Marc Kery and Andy Royle. The first volume appeared early in 2016 (ISBN: 978-0-12-801378-6, <<https://www.mbr-pwrc.usgs.gov/pubanalysis/keryroylebook/>>); the second volume is in preparation and additional functions will be added to this package.

License GPL (>= 3)

NeedsCompilation no

URL <https://www.mbr-pwrc.usgs.gov/pubanalysis/keryroylebook/>,
<https://sites.google.com/site/appliedhierarchicalmodeling/home>

Repository CRAN

Date/Publication 2017-05-19 05:56:57 UTC

R topics documented:

AHMbook-package	2
bigCrossCorr	5
data.fn	6
dragonflies	7

e2dist	8
Finnmark	9
fitstats	10
image_scale	11
issj.sim	12
map.Nmix.resi	14
MHB2014	15
piFuns	16
playRN	18
plot_Nmix_resi	19
ppc.plot	20
sim.fn	21
sim.ldata	22
sim.pdata	23
sim.spatialDS	24
sim.spatialHDS	26
sim3Occ	28
simComm	30
simDCM	33
simDynocc	38
simHDS	41
simHDSg	43
simHDSopen	44
simHDStr	46
simNmix	48
simOcc	51
simOccttd	53
simpleNmix	55
spatial.exp	56
spline.prep	57
SwissSquirrels	58
SwissTits	59
ttdPeregrine	61
wagtail	62
wigglyOcc	63
Index	65

AHMbook-package

Functions and data for the Book 'Applied Hierarchical Modeling in Ecology'

Description

Provides functions and data sets needed to run the code given in the two-volume publication, *Applied Hierarchical Modeling in Ecology: analysis of distribution, abundance and species richness in R and BUGS* by Marc Kéry and Andy Royle, Academic Press (Vol 1, 2016; Vol 2, in preparation).

Details

The functions are listed by chapter below.

Chapter 1

[sim.fn](#) Simulate a Poisson point process and illustrate the relationships between intensity, abundance and occurrence

Chapter 4

[data.fn](#) Simulate count data that are replicated in space and in time according to the binomial mixture model of Royle (2004) (this is for much simpler cases than is possible with `simNmix` below)

Chapter 6

[simNmix](#) Simulate data for binomial and multinomial mixture models under a wide range of conditions

[ppc.plot](#) Plot results from posterior predictive checks in section 6.8, for a fitted model object with JAGS

[plot_Nmix_resi](#) Do diagnostic plots for one Nmix model fitted with all three mixture distributions currently available in unmarked: Poisson, negative binomial and zero-inflated Poisson (Section 6.9.3)

[map.Nmix.resi](#) Produce a map of the residuals from an Nmix model (see Section 6.9.3)

[simpleNmix](#) Simulate data under a simple version of the binomial mixture model, with space for time substitution (section 6.12)

[playRN](#) Play Royle-Nichols model: generate replicated count data under the Nmix model of Royle (2004), then 'degrade' the data to detection/nondetection and fit the RN model using unmarked and estimate site-specific abundance (section 6.13.1)

Chapter 7

[fitstats](#), [fitstats2](#) Calculate fit-statistics used in parboot GOF tests throughout book (Sections 7.5.4, 7.9.3)

[instRemPiFun](#), [crPiFun](#), [crPiFun.Mb](#), [MhPiFun](#) Define the relationship between the multinomial cell probabilities and the underlying detection probability parameters (i.e., a pi function) in various designs (Section 7.8)

Chapter 8

[sim.ldata](#) Simulate data under a non-hierarchical line transect distance sampling model (Section 8.2.3)

[sim.pdata](#) Simulate data under a non-hierarchical point transect (= point count) distance sampling model (Section 8.2.5.1)

[simHDS](#) Simulate data under a hierarchical distance sampling protocol (line or point) (Section 8.5.1)

Chapter 9

- [simHDSg](#) Simulate data under a hierarchical distance sampling (HDS) protocol with groups (Section 9.2.1)
- [simHDStr](#) Simulate data under a time-removal/distance sampling design (Section 9.3.2)
- [simHDSopen](#) Simulate open hierarchical distance sampling data
- [issj.sim](#) Simulate data under the open distance sampling protocol for the Island Scrub Jays (Section 9.7.1)
- [sim.spatialDS](#) Simulate data under a basic spatial distance sampling model (Section 9.8.3)
- [sim.spatialHDS](#) Simulate data under a spatial hierarchical distance sampling model (Section 9.8.5)

Chapter 10

- [simOcc](#) Simulate data under static occupancy models under a wide range of conditions (Section 10.5)
- [sim30cc](#) Simulate data under a static 3-level occupancy model (section 10.10)
- [simOccttd](#) Simulate data under a static time-to-detection occupancy design (section 10.12.1)
- [wigglyOcc](#) Simulate data under a static occupancy model with really wiggly covariate relationships in occupancy and detection probability (section 10.14)
- [spline.prep](#) Prepare input for BUGS model when fitting a spline for a covariate (section 10.14)

Chapter 11

- [simComm](#) Simulate community occupancy or abundance data

Chapter 16

- [simDynocc](#) Simulate data under a dynamic occupancy model under a wide range of conditions

Chapter 17

- [simDCM](#) Simulate detection/nondetection data under a general dynamic community (site-occupancy) model

Chapter 20

- [spatial.exp](#) Generate Gaussian random field with negative exponential correlation

DATA SETS

- [dragonflies](#) Toy data set used in section 3.1
- [Finmark](#) Data from surveys of birds in Finnmark in NE Norway
- [jay](#) The European jay data set is now included in **unmarked**
- [MHB2014](#) Data from the Swiss Breeding Bird Survey for 2014
- [SwissSquirrels](#) Data for red squirrels in Switzerland
- [SwissTits](#) Data for 6 species of tits in Switzerland from 2004 to 2013
- [ttdPeregrine](#) Time-to-detection data for peregrines
- [wagtail](#) Data for yellow wagtails in Netherlands

UTILITY FUNCTIONS

[e2dist](#) Compute Euclidean distances

[image_scale](#) Draw scale for image (from SCR book, introduced in chapter 9)

[bigCrossCorr](#) Report cross-correlations above a given threshold

Author(s)

Marc Kéry, Andy Royle, Mike Meredith

bigCrossCorr

Cross correlations for MCMC output

Description

A wrapper for `coda::crosscorr`, which calculates cross-correlations between variables in Markov Chain Monte Carlo output. When the output has hundreds of parameters, the matrix produced by `crosscorr` is unweildy, and `bigCrossCorr` extracts those greater than a given threshold.

Usage

```
bigCrossCorr(x, big = 0.6, digits = 3)
```

Arguments

`x` an `mcmc` or `mcmc.list` object.
`big` only values below `-big` or above `+big` will be returned
`digits` the number of decimal places to return

Value

A data frame with 2 columns for the names of parameters and a 3rd column with the cross-correlation.

Author(s)

Mike Meredith

See Also

[crosscorr](#) in package `coda`.

data.fn

*Simulate data for binomial mixture model***Description**

Function to simulate point counts replicated at M sites during J occasions. Population closure is assumed for each site. Expected abundance may be affected by elevation (elev), forest cover (forest) and their interaction. Expected detection probability may be affected by elevation, wind speed (wind) and their interaction.

Usage

```
data.fn(M = 267, J = 3, mean.lambda = 2, beta1 = -2, beta2 = 2, beta3 = 1,
        mean.detection = 0.3, alpha1 = 1, alpha2 = -3, alpha3 = 0, show.plot = TRUE)
```

Arguments

M	Number of spatial replicates (sites)
J	Number of temporal replicates (occasions)
mean.lambda	Mean abundance at value 0 of abundance covariates
beta1	Main effect of elevation on abundance
beta2	Main effect of forest cover on abundance
beta3	Interaction effect on abundance of elevation and forest cover
mean.detection	Mean detection prob. at value 0 of detection covariates
alpha1	Main effect of elevation on detection probability
alpha2	Main effect of wind speed on detection probability
alpha3	Interaction effect on detection of elevation and wind speed
show.plot	if TRUE, plots of the data will be displayed; set to FALSE if you are running many simulations

Value

A list with the input arguments and the following additional elements:

elev	Scaled elevation (a vector of length M)
forest	Scaled forest cover (a vector of length M)
wind	Scaled wind speed (an $M \times J$ matrix)
lambda	Expected number of individuals at each site (a vector of length M)
N	Realised number of individuals at each site (a vector of length M)
p	Probability of detection for each survey (an $M \times J$ matrix)
C	The number of detections for each survey (an $M \times J$ matrix)
Ntotal	Total abundance, $\text{sum}(N)$
psi.true	True occupancy in sample
summaxC	Sum of max counts (all sites)
psi.obs	Observed occupancy in sample

Note

The colours used for points in some of the plots indicate different temporal replicates.

Author(s)

Marc Kéry and Andy Royle

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 4.3

Examples

```
# Generate a simulated data set with default arguments and look at the structure:
tmp <- data.fn()
str(tmp)

str(data.fn(J = 2))           # Only 2 surveys
str(data.fn(J = 1))           # No temporal replicate
str(data.fn(M = 1, J = 100))  # No spatial replicates, but 100 counts
str(data.fn(beta3 = 1))      # With interaction elev-wind on p
str(data.fn(M = 267, J = 3, mean.lambda = 2, beta1 = -2, beta2 = 2, beta3 = 1,
  mean.detection = 1))      # No obs. process (i.e., p = 1, perfect detection)
str(data.fn(mean.lambda = 50)) # Really common species
str(data.fn(mean.lambda = 0.05)) # Really rare species
```

dragonflies

Imaginary data for dragonflies

Description

A toy data set with fictional data for 9 dragonflies *Onychogomphus uncatus* from 3 populations in the Spanish Pyrenees.

Usage

```
data(dragonflies)
```

Format

The format is seven vectors of length 9:

pop a factor indicating which population the individual was drawn from.

sex a factor indicating the sex of each individual.

wing wingspan.

body body length.

mites number of mites (ectoparasites) counted.

color proportion of body yellow as opposed to black.

damage the number of wings (out of 4) damaged.

Source

Fictitious data.

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 3.1

Examples

```
data(dragonflies)
lm(wing ~ pop + body)
```

e2dist

Calculates the pair-wise distances between two sets of points

Description

Calculates the Euclidian distance between each of the points defined by the coordinates in 'x' and each of those in 'y'.

Usage

```
e2dist(x, y)
```

Arguments

x a 2-column matrix or data frame with the x and y coordinates of a set of points.
y a 2-column matrix or data frame with the x and y coordinates of a second set of points.

Value

A $nrows(x) \times nrows(y)$ matrix with the pair-wise distances.

Author(s)

J. Andrew Royle

Examples

```
pts1 <- expand.grid(x = 1:5, y = 6:8)
pts2 <- cbind(x=runif(5, 1, 5), y=runif(5, 6, 8))
require(graphics)
plot(pts1)
points(pts2, pch=19, col='red')
e2dist(x=pts1, y=pts2)
```



```
# x and y can be the same:  
e2dist(x=pts2, y=pts2)
```

Finnmark

Data from surveys of birds in Finnmark in NE Norway

Description

A total of 37 plots were placed spanning the existing variation in extent and fragmentation of willow thickets. Each plot was visited 3-5 times in early July each year from 2005 to 2008. Birds observed within 100m during a 15 min period were recorded.

Usage

```
data("Finnmark")
```

Format

Finnmark is a list with 4 elements:

species a data frame with a row for each species and the following columns:

1. species : the English name.
2. latin : the Latin name.
3. assemblage : the guild of the species.

sites a data frame with a row for each plot and the following columns:

1. region : a factor, 3 levels, Ilford, Komag, and Vestre Jakobselv.
2. catchment : a factor, 11 levels.
3. plot : a factor, the alphanumeric ID of each plot.
4. plotnr : the ID number of each plot.
5. area : the percentage of a 400 x 400m quadrat centred on the sampling point covered by tall willow thickets.
6. edge : the length (m) of the edge of the willow thickets in the quadrat.
7. height : the mean height of willows at 4 measuring points.
8. density : a measure of thicket density at 4 measuring points.

counts a sites x replicates x years x species array of counts

timeOfDay a sites x replicates x years array with the time of day that the survey was carried out.

Source

Data generously provided by Jon-Andren Henden.

References

- Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Chapter 17
- Henden J-A, Yoccoz NG, Ims RA, Langeland K (2013) How Spatial Variation in Areal Extent and Configuration of Labile Vegetation States Affect the Riparian Bird Community in Arctic Tundra. PLoS ONE 8(5): e63312.

Examples

```
data(Finmark)
str(Finmark)

# Create the objects needed for the analysis:
# Extract the numeric covariates and standardise
str(Finmark$sites)
scov <- scale(Finmark$sites[, 5:8])
str(scov)

# Convert the 'counts' array to detection/nondetection data:
y <- Finmark$counts > 0
storage.mode(y) <- "integer"
str(y)

# Get the guild information for each species
guild <- Finmark$species$assemblage

# Standardise the timeOfDay and replace NAs with 0
tod <- with(Finmark, (timeOfDay - mean(timeOfDay, na.rm=TRUE))/ sd(timeOfDay, na.rm=TRUE))
tod[is.na(tod)] <- 0
str(tod)
```

fitstats

Functions to return fit statistics

Description

fitstats produces sum of squared errors, Chi-squared statistic and Freeman-Tukey statistic used in parboot GOF tests throughout book, starting with Section 7.5.4. fitstats2 produces the above, plus corresponding statistics based on numbers; see section 7.9.3.

Usage

```
fitstats(fm)
```

```
fitstats2(fm)
```

Arguments

fm A fitted model object as returned by **unmarked**.

Value

For `fitstats`, a named vector of length 3 with sum of squared errors (SSE), Chi-squared statistic (Chisq) and Freeman-Tukey (`freemanTukey`).

For `fitstats2`, a named vector of length 6 with the same information plus corresponding statistics based on numbers(`SSE.n`), (`Chisq.n`) and (`freemanTukey.n`).

Author(s)

Kéry & Royle

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Sections 7.5.4 and 7.9.3.

Examples

```
## TO DO
```

<code>image_scale</code>	<i>Helper function to draw scale for image</i>
--------------------------	--

Description

Plots a scale in the right margin of a plot, typically an image plot.

Usage

```
image_scale(z, col, x, y = NULL, size = NULL, digits = 2, labels = c("breaks", "ranges"))
```

Arguments

<code>z</code>	the values plotted in the image, the <code>z</code> argument to <code>image</code>
<code>col</code>	a vector of colours, use the same as in the image
<code>x</code>	the x coordinate of the left edge of the scale bar; or a list with components <code>x</code> and <code>y</code> , each of length 2, giving the x and y coordinates of the edges of the scale bar.
<code>y</code>	the y coordinate of the bottom edge of the scale bar, ignored if <code>x</code> is a list
<code>size</code>	the size of the boxes making up the scale bar, a length 2 vector with the width and height, or a scalar, in which case width = height; ignored if <code>x</code> is a list
<code>digits</code>	the number of decimal places to display
<code>labels</code>	if "breaks", the dividing lines in the scale bar are labelled, if "ranges", each box is labelled with its range; may be abbreviated

Value

None, used for its plotting side effect.

Note

This function appears in the text as `image.scale`; renamed to avoid confusion with generic `image` functions.

Author(s)

Borrowed from the **SCRbook** package.

References

Royle, J.A., Chandler, R.B., Sollmann, R., & Gardner, B. (2014) *Spatial capture-recapture* Elsevier.

Examples

```
# uses the built-in volcano data set
require(grDevices) # for colours
require(graphics)
par(mar = c(3,3,3,6)) # make the right margin wide enough
image(t(volcano)[ncol(volcano):1,], col=terrain.colors(12))
image_scale(volcano, col=terrain.colors(12))
# Try placing the scale bar on the left
par(mar = c(3,8,3,1)) # make the left margin wide enough
image(t(volcano)[ncol(volcano):1,], col=terrain.colors(12))
image_scale(volcano, col=terrain.colors(12), x= -0.28, digits=0)
# Trial and error needed to get the x value right.
```

 issj.sim

Simulate open distance sampling data for the Island Scrub Jays

Description

Function to simulate open distance sampling data for the Island Scrub Jays, based on Sollmann et al (2015).

Usage

```
issj.sim(B, db, lam, sigma, phi, gamma, npoints, nyrs, nbsize = -1.02)
```

Arguments

B	Radius of the circle sampled; a site is a circle of radius B around a point.
db	Distance categories; a vector of cut points from 0 to B inclusive.
lam	Expected abundance per site, a vector of length <code>nsites</code> or an <code>nsites</code> x 1 matrix.
sigma	Scale parameter of the half-normal detection function at each site, a vector of length <code>nsites</code> or an <code>nsites</code> x 1 matrix.
phi	Survival probability

gamma	Recruitment rate
npoints	Number of sites where point counts are conducted.
nyrs	Number of years
nbsize	Size parameter for the negative binomial distribution used to generate individual counts per site for year 1.

Value

A list with the following elements:

NcList	A list with one element per year, with distances of all animals from the point.
detList	A list with one element per year, a npoints x nbands matrix with the number of animals detected in each distance category for each point.
N	The (true) number of animals at each point for each year, a nsites x nyears matrix.
cell	The site IDs where point counts are conducted.
y	npoints x nyears matrix of total detections
dclass	a vector with the distance class for each animal detected
site	a corresponding vector with the site for each animal detected
nsite	the number of sites in the study area
lam, phi, gamma, sigma	the values of the corresponding arguments

Author(s)

Based on Sollmann et al (2015)

References

- Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 9.7.1
- Sollmann, R., Gardner, B., Chandler, R.B., Royle, J.A., Sillett, T.S. (2015) An open population hierarchical distance sampling model. *Ecology* 96, 325-331.

Examples

```
# A toy example with just 20 sites
set.seed(2015)
tmp <- issj.sim(B = 300,
  db = c(0,50, 100, 150, 200, 250, 300),
  lam = c(3.01, 7.42, 20.51, 1.60, 0.42, 3.42, 8.24, 0.66, 0.32, 0.39, 0.46, 0.52,
    0.63, 0.36, 4.93, 0.47, 2.07, 0.42, 0.48, 0.47),
  sigma = c(110, 91, 70, 114, 135, 101, 88, 130, 133, 134, 134, 135, 131, 135, 100,
    135, 110, 135, 134, 135),
  phi = 0.6, gamma = 0.35,
  npoints = 15, nyrs = 4)
str(tmp)
# Compare the number detected with the true numbers present
with(tmp, cbind(y, N[cell, ]))
```

map.Nmix.resi	<i>Mapping of residuals</i>
---------------	-----------------------------

Description

Produces a map of the mean residuals from an N-mixture model fit by function `pcount` in **unmarked**. Used in Section 6.9.3 to produce maps of Switzerland with the residuals for each site.

Usage

```
map.Nmix.resi(fm, x, y)
```

Arguments

<code>fm</code>	the fitted model object
<code>x</code>	x coordinates of each site
<code>y</code>	y coordinates of each site

Value

None. Used for its plotting side effects.

Note

In previous versions, the defaults were `x = tits$coordx` and `y = tits$coordy`, but those defaults only worked if the data object `tits` was in the workspace. To run the code on page 263 of AHM1, you now need to specify the coordinates, eg, `map.Nmix.resi(fm5, x = tits$coordx, y = tits$coordy)`.

Author(s)

Marc Kéry

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 6.9.3

Examples

```
# Not possible as would require a fitted model object. See section 6.9.3 of AHM1.
```

MHB2014

*Data from the Swiss Breeding Bird Survey 2014***Description**

The Swiss breeding bird survey ("Monitoring Häufige Brutvögel" MHB) has monitored the populations of 150 common species since 1999. The MHB sample consists of 267 1-km squares that are laid out as a grid across Switzerland. Fieldwork is conducted by about 200 skilled birdwatchers, most of them volunteers. Avian populations are monitored using a simplified territory mapping protocol, where each square is surveyed up to three times during the breeding season (only twice above the tree line). Surveys are conducted along a transect that does not change over the years.

The list MHB2014 has the data for 2014.

Usage

```
data("MHB2014")
```

Format

MHB2014 is a list with 5 elements:

species a data frame with row for 158 species, including 13 species not recorded in 2014, and the following columns:

1. `specid` : a numeric species ID based on phylogeny.
2. `latabb` : a 6-letter abbreviation of the Latin name.
3. `engname` : the English name.
4. `latname` : the Latin name.
5. `body.length` : body length in cm.
6. `body.mass` : body mass in g.
7. `wing.span` : wing span in cm.

sites a data frame with rows for 267 1x1 km quadrat, including 1 quadrat not surveyed in 2014, and the following columns:

1. `siteID` : an alphanumeric site identifier.
2. `coordx` : the x coordinate of the centre of the quadrat; the coordinate reference system intentionally not specified.
3. `coorxy` : the y coordinate of the centre of the quadrat.
4. `elev` : the mean elevation of the quadrat, m.
5. `rlength` : the length of the route walked in the quadrat, km.
6. `nsurvey` : the number of replicate surveys *planned* in the quadrat; above the tree-line 2, otherwise 3.
7. `forest` : percentage forest cover.
8. `obs14` : identifying number of the observer.

counts a sites x replicates x species array of counts

date a sites x replicates matrix with Julian dates of the surveys, 1 April = 1

dur a sites x replicates matrix with the duration of each survey, mins

Note

Section 11.3 of the book has code to read in data from a CSV file, "MHB_2014.csv". This is a huge file, because the site data are repeated for all 158 species and the species data are repeated for all 267 sites. The MHB2014 list has all the same data in a more compact format. See Examples for ways to generate the objects used in the book from the list.

Source

Swiss Ornithological Institute

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 11.3

Examples

```
data(MHB2014)
str(MHB2014)

# Create the objects at foot of p.644:
( nsite <- nrow(MHB2014$sites) ) # number of sites in Swiss MHB
nrep <- 3 # maximum number of replicate surveys per season
( nspec <- nrow(MHB2014$species) ) # 158 species occur in the 2014 data
# Check the dimensions of the 'count' array:
dim(MHB2014$count) == c(nsite, nrep, nspec)

# Create the detection/nondetection matrix 'y':
y <- MHB2014$count > 0 # this is logical, convert to integer
storage.mode(y) <- "integer" # don't use 'as.integer', that strips out dimensions and names
str(y)

# Pull out and check the data for common chaffinch, p.645:
head(tmp <- y[, , "Common Chaffinch"])
tail(tmp)
```

Description

Compute the cell probabilities used in the multinomial-Poisson models `multinomPois` and `gmultmix` in package **unmarked**. See [piFuns](#).

Usage

instRemPiFun(p)

crPiFun(p)

crPiFun.Mb(p)

MhPiFun(p)

Arguments

p matrix of detection probabilities at each site for each observation

Details

instRemPiFun defines the relationship between the multinomial cell probabilities and the underlying detection probability parameters (i.e., a pi function) in a removal design with unequal length of sampling periods.

crPiFun defines a pi function for capture-recapture design with 3 surveys (defined in Section 7.8.5). NOTE that this is not the same as the custom crPiFun defined in Section 7.9.1.

crPiFun.Mb defines a pi function for capture-recapture design with 3 surveys and behavioural response (introduced in Section 7.8.2)

MhPiFun defines a pi function for model with individual detection heterogeneity (introduced in Section 7.8.3)

Value

TODO

Author(s)

Kéry & Royle

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Chapter 7

Examples

TODO

 playRN

Function to play Royle-Nichols model

Description

Function generates replicated count data under the Nmix model of Royle (2004), then 'degrades' the data to detection/nondetection and fits the RN model (Royle & Nichols 2003) using **unmarked** and estimates site-specific abundance.

Usage

```
playRN(M = 267, J = 3, mean.abundance = 1, mean.detection = 0.3, show.plot = TRUE)
```

Arguments

M	The number of sites.
J	The number of visits to each site.
mean.abundance	Expected abundance at each site.
mean.detection	Expected detection at each survey at each site.
show.plot	choose whether to show plots or not. Set to FALSE when using function in simulations.

Value

A list with the following elements:

nsite	The number of sites, equal to M in the input
nvisit	The number of visits, equal to J in the input
coef	A named vector of coefficients for the linear models for expected number and detection probability
slope	Slope of the regression of the estimated number on the true number; 1 if the model is perfect

Author(s)

Kéry & Royle

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 6.13.1

Examples

```
# Run a simulation with the default arguments and look at the results:
playRN()

# Execute the function using various settings
playRN(M = 100, J = 3, mean.abundance = 0.1) # Increasing abundance
playRN(M = 100, J = 3, mean.abundance = 1)
playRN(M = 100, J = 3, mean.abundance = 5)
playRN(M = 100, J = 3, mean.detection = 0.3) # Increasing detection
playRN(M = 100, J = 3, mean.detection = 0.5)
playRN(M = 100, J = 3, mean.detection = 0.7)
playRN(M = 100, J = 20) # More visits
playRN(M = 1000, J = 3) # More sites
```

plot_Nmix_resi *Produce some residual plots*

Description

Function does diagnostic plots for one Nmix model fitted with all three mixture distributions currently available in package **unmarked**: Poisson, negative binomial and zero-inflated Poisson. For each, fitted values vs. observed data and residuals vs. fitted values are plotted.

Usage

```
plot_Nmix_resi(fmP, fmNB, fmZIP)
```

Arguments

fmP	Fitted model object for Poisson distribution.
fmNB	Fitted model object for negative binomial distribution.
fmZIP	Fitted model object for zero-inflated Poisson distribution.

Value

None, used for its plotting effect.

Note

This function appears in the text as `plot.Nmix.resi`; renamed to avoid confusion with generic `plot` functions.

Author(s)

Marc Kéry

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 6.9.3

Examples

```
##
```

ppc.plot

Plot results from posterior predictive check

Description

Function plots results from posterior predictive check in AHM section 6.8 for a fitted model object with JAGS.

Usage

```
ppc.plot(fm)
```

Arguments

fm The fitted model object

Value

None, used for its plotting side effect.

Author(s)

Marc Kéry,

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 6.8

Examples

```
## We would need a fitted model object to demonstrate this, but that takes a long time to run
```

sim.fn

*Simulate a Poisson point process***Description**

Simulates animal or plant locations in space according to a homogenous Poisson process. This process is characterized by the intensity, which is the average number of points per unit area. The resulting point pattern is then discretized to obtain abundance data and presence/absence (or occurrence) data. The discretization of space is achieved by choosing the cell size. It is used in Section 1.1 to help to understand the relationship between point patterns, abundance data and occurrence data (also called presence/absence or distribution data).

Usage

```
sim.fn(quad.size = 10, cell.size = 1, intensity = 1, show.plot = TRUE)
```

Arguments

quad.size	The length of each side of the quadrat (in arbitrary units)
cell.size	The length of each side of the cells into which the quadrat is divided. The ratio of quad.size to cell.size must be an integer.
intensity	The average number of points (animals or plants) per unit area.
show.plot	If TRUE, the results are plotted. Set to FALSE when running many simulations.

Value

A list with the values of the arguments and the following additional elements:

exp.N	Expected population size in quadrat
breaks	boundaries of grid cells
n.cell	Number of cells in the quadrat
mid.pt	Cell mid-points
M	Realized population size in quadrat
u1	x coordinate of each individual
u2	y coordinate of each individual
N	The number of individuals in each cell (a vector of length n.cell)
z	Presence/absence (1/0) in each cell (a vector of length n.cell)
psi	Proportion of cells occupied, ie, the species is present.

Author(s)

Marc Kéry and Andy Royle

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 1.1

Examples

```
# Generate a simulated data set with default arguments and look at the structure:
tmp <- sim.fn()
str(tmp)

# Effect of grain size of study on abundance and occupancy (intensity constant)
tmp <- sim.fn(quad.size = 10, cell.size = 1, intensity = 0.5)
tmp <- sim.fn(quad.size = 10, cell.size = 2, intensity = 0.5)
tmp <- sim.fn(quad.size = 10, cell.size = 5, intensity = 0.5)
tmp <- sim.fn(quad.size = 10, cell.size = 10, intensity = 0.5)

# Effect of intensity of point pattern (intensity) on abundance and occupancy
tmp <- sim.fn(intensity = 0.1) # choose default quad.size = 10, cell.size = 1
tmp <- sim.fn(intensity = 1)
tmp <- sim.fn(intensity = 5)
tmp <- sim.fn(intensity = 10)
```

sim.ldata

Simulation of distance sampling data.

Description

Simulates non-hierarchical line transect data under conventional distance sampling (CDS). It subjects N individuals to sampling, and then retains the value of distance from transect only for individuals that are captured.

Usage

```
sim.ldata(N = 200, sigma = 30, show.plot = TRUE)
```

Arguments

N	number of individuals along transect with distance $u(-100, 100)$
sigma	scale parameter of half-normal detection function
show.plot	choose whether to show plots or not. Set to FALSE when using function in simulations.

Value

A list with the following elements:

N	the number of individuals along the transect
sigma	scale parameter of half-normal detection function
xall	distance from the transect line for all N individuals
x	absolute distance from the transect line for those individuals detected

Author(s)

Kéry & Royle

ReferencesKéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 8.2.3**Examples**

```
# Simulate a data set with the default arguments and look at the structure of the output:
tmp <- sim.ldata()
str(tmp)
```

sim.pdata

*Simulate non-hierarchical point transect (= point count) data***Description**

Function simulates coordinates of individuals on a square with a count location at the center point.

Usage

```
sim.pdata(N = 1000, sigma = 1, B = 3, keep.all = FALSE, show.plot = TRUE)
```

Arguments

N	total population size in the square
sigma	scale of half-normal detection function
B	circle radius; the data are simulated over a square of side $2 * B$, but individuals outside the circle of radius B are not detected.
keep.all	if TRUE, the data for all individuals are returned; if FALSE, only for individuals detected.
show.plot	choose whether to show plots or not. Set to FALSE when using function in simulations.

Value

A list with the following elements:

N	total population size in the square
sigma	scale of half-normal detection function
B	circle radius
u1, u2	the x and y coordinates of each of the individuals
d	the distance of each individual from the centre of the circle
y	a 0/1 indicator showing whether each individual is detected or not, a vector of length N
N.real	the realised number of individuals within the circle of radius B

Author(s)

Kéry & Royle

ReferencesKéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 8.2.5.1**Examples**

```
# Simulate a data set with the default arguments and look at the structure of the output:
tmp <- sim.pdata()
str(tmp)
```

sim.spatialDS

Simulates data for a basic spatial distance sampling model

Description

Generates data with the following steps:

1. Simulate a spatially correlated habitat covariate (x) over a grid of pixels covering a square.
2. Distribute the population of N individuals over the square with probability of location in a pixel related to the covariate.
3. Decide which individuals are detected using a distance sampling model with an observer at the centre of the square, with either a half normal or a logit detection function. (Note that all the individuals in the square can be detected.)
4. If `keep.all = FALSE`, return the locations of only the individuals detected.

Usage

```
sim.spatialDS(N = 1000, beta = 1, sigma = 1, keep.all = FALSE, B = 3,
  model=c("logit", "halfnorm"), lambda = B/3, useHabitat, show.plot=TRUE)
```

Arguments

N	total population size in the square
beta	coefficient of SOMETHING on spatial covariate x
sigma	scale parameter of detection function
keep.all	if TRUE, the data for all individuals are returned; if FALSE, only for individuals detected.
B	distance from the observer to the side of the square. This is usually set so that the probability of detection of individuals outside the square is negligible, eg, $B = 3 \times \text{sigma}$.
model	The detection function used, can be "logit" or "halfnorm": see Details.
lambda	The scale parameter for the spatially autocorrelated Habitat covariate.

useHabitat	If the output from a previous simulation is provided, the same Habitat covariate will be used (and lambda will be ignored).
show.plot	choose whether to show plots or not. Set to FALSE when using function in simulations.

Details

The "logit" detection function is $2 * \text{plogis}(-d^2 / (2 * \sigma^2))$, which corresponds to the detection model implemented in `unmarked::pcount.sphDS`.

Value

A list with the values of the input arguments and the following additional elements:

u1	x coordinate of each animal
u2	y coordinate of each animal
d	distance of each animal from the centre of the circle
pixel.id	the pixel in which each animal is located, the row number in <code>grid</code>
y	indicator of detection of each animal, a vector of length N
N.real	the number of animals inside the circle of radius B
Habitat	Value of the spatially correlated habitat covariate, a 900 x 1 matrix
grid	Coordinates of the centres of each pixel, a dataframe with 900 rows and 2 columns

If `keep.all = FALSE` (the default), only the animals detected are included in `u1`, `u2`, `d`, `pixel.id`.

Note

Kéry & Royle (2016, p.535 and discussion p.540) and earlier versions of AHMbook included a hazard rate detection function. This is problematic because the detection probability at distance zero is less than 1 ($p(0) < 1$) and should not be used. It is replaced here with the logit detection function, which does have $p(0) = 1$.

Author(s)

Kéry & Royle

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 9.8.3

Examples

```
# Generate data with the default arguments and look at the structure:
tmp <- sim.spatialDS()
str(tmp)

# Generate data with model = "logit" and analyse the data with unmarked::pcount.spatialHDS
set.seed(1234)
```

```

tmp <- sim.spatialDS(model="logit")
# Plot shows a large area of good habitat west of the observer with many animals detected
str(tmp) # 272 animals detected out of 850 inside the circle (N.real)

# Get the count of animals detected in each pixel
pixel.count <- tabulate(tmp$pixel.id, nbins=nrow(tmp$grid))
# Centre the Habitat covariate
Habitat <- tmp$Habitat - mean(tmp$Habitat)
# Create a detection covariate: distance between observer and pixel center
dist <- with(tmp, sqrt((grid[,1]-B)^2 + (grid[,2]-B)^2))
# Construct an unmarkedFrame
umf <- unmarkedFramePCount(y=cbind(pixel.count),
  siteCovs=data.frame(dist=dist, Habitat=Habitat))
summary(umf)

# Fit some models
(fm0 <- pcount.spHDS(~ -1 + I(dist^2) ~ 1, umf, K=20))
(fm1 <- pcount.spHDS(~ -1 + I(dist^2) ~ Habitat, umf, K=20))
# The model with Habitat has much lower AIC
# Get an estimate of the total population in the square (true is N = 1000)
sum(predict(fm1, type='state')[, 1])

```

sim.spatialHDS

Simulates data for a hierarchical spatial distance sampling model

Description

Generates data for distance sampling from spatially-replicated point transects, with density dependent on a spatially correlated habitat covariate. For each point count, the procedure is:

1. Simulate the habitat covariate over a grid of pixels covering a square.
2. Distribute the population of individuals over the square with probability of location in a pixel related to the covariate.
3. Decide which individuals are detected using a distance sampling model with an observer at the centre of the square, with a half normal detection function. (Note that individuals outside the circle of radius B can be detected.)

The locations and detection status of individuals at all sites are collated and returned, except for individuals at sites when none are detected.

Usage

```

sim.spatialHDS(lam0 = 4, sigma = 1.5, B = 3, nsites = 100,
  beta1 = 1, npix = 20, show.plot=3)

```

Arguments

lam0 the expected number of individuals in the square of side = 2*B if the habitat covariate had no effect (ie, beta1 = 0).

sigma	scale parameter of the half-normal detection function
B	distance from the observer to the side of the square. This is usually set so that the probability of detection of individuals outside the square is negligible, eg, $B = 3 * \text{sigma}$.
nsites	number of sites
beta1	the size of the effect of the habitat covariate on the number of individuals in a pixel.
npix	the number of pixels along each dimension of the square: the entire grid has $\text{npix} \times \text{npix}$ pixels.
show.plot	the number of sites for which plots should be displayed.

Value

A list with the following components:

data	a matrix with columns for siteID, the coordinates of each individual (u_1 , u_2), distance from the centre of the square (d) and detection status, 0/1 (y), and rows for each individual, except that individuals at sites where no individuals were detected are NOT included, and those sites are represented by NAs
B	the radius of the circle; the data are simulated over a square of side $2 * B$
Habitat	a matrix with the values of the Habitat covariate for each pixel at each site
grid	a data frame with the coordinates of each pixel (same for all sites)
N	the realised number of individuals in the square at each site
nsites	the number sites

Author(s)

Andy Royle

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 9.8.5

Examples

```
# Generate data with the default arguments and look at the structure:
tmp <- sim.spatialHDS()
str(tmp)
```

sim3Occ

*Simulate data for static 3-level occupancy models***Description**

Function generates 3-level occupancy data with possibility of site-specific random variation at every level, "time effects" at the middle and the lower levels and effects of one distinct covariate at each level.

Usage

```
sim3Occ(nunit = 100, nsubunit = 5, nrep = 3,
        mean.psi = 0.8, beta.Xpsi = 1, sd.logit.psi = 0,
        mean.theta = 0.6, theta.time.range = c(-1, 1), beta.Xtheta = 1, sd.logit.theta = 0,
        mean.p = 0.4, p.time.range = c(-2, 2), beta.Xp = -1, sd.logit.p = 0,
        show.plot=TRUE)
```

Arguments

nunit	Number of main units (large quadrats)
nsubunit	Number of subunits (nested subsamples within each main unit)
nrep	Number of replicate surveys in every subunit
mean.psi	Mean large-scale, unit-level occupancy probability (psi)
beta.Xpsi	effect on psi of covariate A (at main unit level)
sd.logit.psi	SD of logit(psi), unstructured site variation in psi
mean.theta	Mean small-scale (subunit) occupancy probability (theta)
theta.time.range	range of theta 'intercepts' for subunits
beta.Xtheta	effect on theta of covariate B (at subunit level)
sd.logit.theta	SD of logit(theta), unstructured site variation in theta
mean.p	Mean per-survey detection probability
p.time.range	range of p 'intercepts' for replicates
beta.Xp	effect on p of covariate C (unit by subunit by replicate)
sd.logit.p	SD of logit(p)
show.plot	choose whether to show plots or not. Set to FALSE when using function in simulations.

Value

A list with the values of the input arguments and the following additional elements:

theta.time.effect
 Simulated time effect on theta, a vector of length nsubunit

p.time.effect	Simulated time effect on p, a vector of length nrep
p	Detection probability, a nunit x nsubunit x nrep array
z	Occupancy indicator for main units, a nunit x 1 array of 0/1
a	Occupancy indicator for subunits, a nunit x nsubunit array of 0/1
y	Detection array, a nunit x nsubunit x nrep array of 0/1
sum.z	True number of occupied main units
obs.sum.z	Observed number of occupied main units
sum.z.a	Number of units with >=1 occupied, surveyed subunit
covA	Simulated covariate A, a vector of length nunit
covB	Simulated covariate B, a nunit x nsubunit matrix
covC	Simulated covariate C, a nunit x nsubunit x nrep array

Author(s)

Marc Kéry

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 10.10

Examples

```
# Generate data with the default arguments and look at the structure:
tmp <- sim3occ()
str(tmp)

# 'Null' model (model 1)
str(data <- sim3occ(nunit = 100, nsubunit = 5, nrep = 3, mean.psi = 0.8,
  beta.Xpsi = 0, sd.logit.psi = 0, mean.theta = 0.6, theta.time.range = c(0, 0),
  beta.Xtheta = 0, sd.logit.theta = 0, mean.p = 0.4, p.time.range = c(0,0),
  beta.Xp = 0, sd.logit.p = 0))

# No covariate effects, no random variability (model 2)
str(data <- sim3occ(nunit = 100, nsubunit = 5, nrep = 3, mean.psi = 0.8,
  beta.Xpsi = 0, sd.logit.psi = 0, mean.theta = 0.6, theta.time.range = c(-1, 1),
  beta.Xtheta = 0, sd.logit.theta = 0, mean.p = 0.4, p.time.range = c(-2,2),
  beta.Xp = 0, sd.logit.p = 0))

# All covariate effects, but no random variability (model 3)
str(data <- sim3occ(nunit = 100, nsubunit = 5, nrep = 3, mean.psi = 0.8,
  beta.Xpsi = 1, sd.logit.psi = 0, mean.theta = 0.6, theta.time.range = c(-1, 1),
  beta.Xtheta = 1, sd.logit.theta = 0, mean.p = 0.4, p.time.range = c(-2,2),
  beta.Xp = -1, sd.logit.p = 0))

# Most complex model with all effects allowed for by sim function (model 4)
str(data <- sim3occ(nunit = 100, nsubunit = 5, nrep = 3, mean.psi = 0.8,
  beta.Xpsi = 1, sd.logit.psi = 1, mean.theta = 0.6, theta.time.range = c(-1, 1),
  beta.Xtheta = 1, sd.logit.theta = 1, mean.p = 0.4, p.time.range = c(-2,2),
  beta.Xp = -1, sd.logit.p = 1))
```

simComm

*Simulate community occupancy or community abundance data***Description**

Function to simulate community occupancy or community abundance data with random species effects for psi/lambda and p (both including effects of one covariate, 'habitat' for psi/lambda and 'wind speed' for p) (introduced in Section 11.2)

Usage

```
simComm(type = c("det/nondet", "counts"), nsite = 30, nrep = 3, nspec = 100,
        mean.psi = 0.25, sig.lpsi = 1, mu.beta.lpsi = 0, sig.beta.lpsi = 0,
        mean.lambda = 2, sig.loglam = 1, mu.beta.loglam = 1, sig.beta.loglam = 1,
        mean.p = 0.25, sig.lp = 1, mu.beta.lp = 0, sig.beta.lp = 0, show.plot = TRUE)
```

Arguments

type	choose whether you want to simulate detection/nondetection ("det/nondet") data or count data ("counts").
nsite	number of sites
nrep	number of replicate samples (occasions or repeated measurements)
nspec	total number of species in the area that is sampled by these sites (regional species pool)
mean.psi	community mean of occupancy probability over all species in community (probability scale)
sig.lpsi	community standard deviation of logit(occupancy probability intercept)
mu.beta.lpsi	community mean of the effects of 'habitat' covariate on occupancy probability
sig.beta.lpsi	community standard deviation of the effects of 'habitat' covariate on occupancy probability
mean.lambda	community mean of expected abundance over all species in superpopulation
sig.loglam	community standard deviation of log(lambda intercept)
mu.beta.loglam	community mean of the effects of 'habitat' covariate on log(lambda)
sig.beta.loglam	community standard deviation of the effects of 'habitat' covariate on expected abundance
mean.p	community mean of detection probability over all species in superpopulation (probability scale)
sig.lp	community standard deviation of logit(detection probability intercept)
mu.beta.lp	community mean of the effects of 'wind' covariate on detection probability
sig.beta.lp	community standard deviation of the effects of 'wind' covariate on detection probability
show.plot	choose whether to show plots or not. Set to FALSE when using function in simulations.

Details

Function simulates data from repeated sampling of a metacommunity (or spatially structured community) according the model of Dorazio & Royle (JASA, 2005) for type = "det/nondet" (this is the default) or under the model of Yamaura et al. (2012) for type = "counts".

Occupancy probability (psi) or expected abundance (lambda) can be made dependent on a continuous site covariate 'habitat', while detection probability can be made dependent an observational covariate 'wind'. Both intercept and slope of the two log- or logistic regressions (for occupancy or expected abundance, respectively, and for detection) are simulated as draws from a normal distribution with mean and standard deviation that can be selected using function arguments.

Specifically, the data are simulated under the following linear models:

(1) for type = "det/nondet" (i.e., community occupancy)

```
(occupancy (psi) and detection (p) for site i, replicate j and species k)
psi[i,k] <- plogis(beta0[k] + beta1[k] * habitat[i]           Occupancy
p[i,j,k] <- plogis(alpha0[k] + alpha1[k] * wind[i,j]       Detection
```

(2) for type = "counts" (i.e., community count)

```
(exp. abundance (lambda) and detection (p) for site i, rep. j and species k)
lambda[i,k] <- exp(beta0[k] + beta1[k] * habitat[i]         E(N)
p[i,j,k] <- plogis(alpha0[k] + alpha1[k] * wind[i,j]       Detection
```

Species-specific heterogeneity in intercepts and slopes is modelled by up to four independent normal distributions (note: no correlation between the intercepts as in Dorazio et al. (2006) or Kéry & Royle (2008))

(1) for type = "det/nondet" (i.e., community occupancy)

```
beta0 ~ dnorm(logit(mean.psi), sig.lpsi)   Mean and SD of normal distribution
beta1 ~ dnorm(mu.beta.lpsi, sig.beta.lpsi)
alpha0 ~ dnorm(logit(mean.p), sig.lp)
alpha1 ~ dnorm(mu.beta.lp, sig.beta.lp)
```

(2) for type = "counts" (i.e., community count)

```
beta0 ~ dnorm(log(mean.lambda), sig.loglam)   Mean and SD of normal distribution
beta1 ~ dnorm(mu.beta.loglam, sig.beta.loglam)
alpha0 ~ dnorm(logit(mean.p), sig.lp)
alpha1 ~ dnorm(mu.beta.lp, sig.beta.lp)
```

Value

A list with the arguments supplied and the following additional elements:

(1) for type = "det/nondet" (i.e., community occupancy)

```
psi           psi[i, k] occupancy probability for site i and species k
p             p[i, j, k] detection probability for site i, occasion j and species k
```

z	z[i, k] true presence/absence for site i and species k
z.obs	z.obs[i, k] observed presence/absence for site i and species k
y.all	y.all[i, j, k] detection/nondetection data for site i, occasion j and species k for all species
y.obs	y.obs[i, j, k] detection/nondetection data for site i, occasion j and species k for observed species
y.sum.all	detection frequency for all species
y.sum.obs	detection frequency for observed species
Ntotal.fs	finite sample (or conditional) species richness
Ntotal.obs	observed species richness
S.true	true number of species occurring at each site
S.obs	observed number of species occurring at each site

(2) for type = "counts" (i.e., community count)

lambda	lambda[i, k] expected number of individuals for site i and species k
p	p[i, j, k] detection probability for site i, occasion j and species k
N	N[i, k] true number present for site i and species k
y.all	y.all[i, j, k] number observed for site i, occasion j and species k for all species
y.obs	y.obs[i, j, k] number observed for site i, occasion j and species k for observed species
y.max.obs	y.max.obs[i, k] maximum number observed for site i and species k
Ntotal.fs	finite sample (or conditional) species richness
Ntotal.obs	observed species richness

Author(s)

Community occupancy model code partly based on code by Richard Chandler.

References

- Dorazio et al (2006) Estimating species richness and accumulation by modeling species occurrence and detectability. *Ecology* 87, 842-854
- Kéry & Royle (2008) Hierarchical Bayes estimation of species richness and occupancy in spatially replicated surveys. *J. Appl. Ecol.* 45, 589-598
- Dorazio & Royle (2005) Estimating size and composition of biological communities by modeling the occurrence of species. *J American Statistical Association*, 100, 389-398.
- Yamamura et al. (2012) Biodiversity of man-made open habitats in an underused country: a class of multispecies abundance models for count data. *Biodivers. Conserv.* 21, 1365-1380.
- Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 11.2

Examples

```
# Default arguments:
str(simComm())

# Some possibly interesting settings of the function
data <- simComm(nsite = 267, nspec = 190, mean.psi = 0.25, sig.lpsi = 2,
  mean.p = 0.12, sig.lp = 2) # similar to Swiss MHB
data <- simComm(mean.psi = 1)      # all species occur at every site
data <- simComm(mean.p = 1)      # no measurement error (perfect detection)

# Effect of spatial sample size (nsite) on species richness in sample (Ntotal.fs)
data <- simComm(nsite=50, nspec = 200) # 1-3 are usually missed in sample
data <- simComm(nsite=30, nspec = 200) # 4-6 usually missed
data <- simComm(nsite=10, nspec = 200) # around 30 typically missed
```

simDCM	<i>Simulate detection/nondetection data under a general dynamic community (site-occupancy) model</i>
--------	--

Description

Function to simulate detection/nondetection data under a general dynamic community (= dynamic, multi-species site-occupancy) model, including:

- * annual variation in the probabilities of patch persistence, colonization and detection is specified by the bounds of a uniform distribution.
- * species heterogeneity around the means is specified by the SD of a normal distribution and expressed on the logit scale
- * one covariate is allowed to a parameter (site covariate for psi1, site-year covariate for phi and gamma and site-year-rep covariate for p). Each covariate is allowed to differ among species again according to a logit-normal model of heterogeneity.
- * additional detection heterogeneity at the site- or the occasion level, with the possibility of a temporal trend in this heterogeneity over years. E.g., an annual trend in detection heterogeneity at the site or the occasion level is specified by the value in the first and the last year. Hence, `range.sd.site = c(0, 1)` will result in a linear trend in the magnitude of site-level heterogeneity in detection from 0 in the first year to 1 in the last year, with interpolation for the years in between.
- * additional detection heterogeneity that among occasions according to a quadratic effect of occasion number (to model phenology of an insect species for instance).

These last two types of detection heterogeneity are not (yet) allowed to be species-specific.

Usage

```
simDCM(nspec = 50, nsite = 100, nrep = 3, nyear = 10,
  mean.psi1 = 0.4, sig.lpsi1 = 1, mu.beta.lpsi1 = 0, sig.beta.lpsi1 = 0,
  range.mean.phi = c(0.8, 0.8), sig.lphi = 1, mu.beta.lphi = 0,
  sig.beta.lphi = 0, range.mean.gamma = c(0.2, 0.2), sig.lgamma = 1,
```

```
mu.beta.lgamma = 0, sig.beta.lgamma = 0, range.mean.p = c(0.5, 0.5),
sig.lp = 1, mu.beta.lp = 0, sig.beta.lp = 0, range.beta1.season = c(0, 0),
range.beta2.season = c(0, 0), range.sd.site = c(0, 0),
range.sd.survey = c(0, 0), show.plot = TRUE)
```

Arguments

<code>nspec</code>	number of species (typically called N in AHM book)
<code>nsite</code>	number of sites (M)
<code>nrep</code>	number of replicate occasions within a year (J)
<code>nyear</code>	number of years (T)
<code>mean.psi1</code>	average (across all species in the community) of the intercept of occupancy probability in first year
<code>sig.lpsi1</code>	sd of the normal distribution from which species-specific occupancy intercepts are drawn (centered on $\text{logit}(\text{mean.psi1})$), on logit scale
<code>mu.beta.lpsi1</code>	community mean of the coefficients of the covariate in probabilities of initial occupancy: the probability-scale mean of the normal distribution from which the species-specific coefficients are drawn.
<code>sig.beta.lpsi1</code>	sd of the normal distribution from which species-specific slopes are drawn (centered on mu.beta.lpsi1)
<code>range.mean.phi</code>	bounds of uniform distribution from which the average (across species) annual intercept of persistence is drawn
<code>sig.lphi</code>	sd of the normal distribution from which species-specific persistence intercepts are drawn (centered on $\text{logit}(\text{mean.phi})$, which are year-specific), on logit scale
<code>mu.beta.lphi</code>	community mean of the coefficients of the covariate in probabilities of persistence: the probability-scale mean of the normal distribution from which the species-specific coefficients are drawn.
<code>sig.beta.lphi</code>	sd of the normal distribution from which species-specific persistence slopes are drawn (centered on mu.beta.lphi)
<code>range.mean.gamma</code>	bounds of uniform distribution from which the average (across species) annual intercept of colonisation is drawn
<code>sig.lgamma</code>	sd of the normal distribution from which species-specific colonization intercepts are drawn (centered on $\text{logit}(\text{mean.gamma})$, which are year-specific), on logit scale
<code>mu.beta.lgamma</code>	community mean of the coefficients of the covariate in probabilities of colonization: the probability-scale mean of the normal distribution from which the species-specific coefficients are drawn.
<code>sig.beta.lgamma</code>	sd of the normal distribution from which species-specific colonization slopes are drawn (centered on mu.beta.lgamma)
<code>range.mean.p</code>	bounds of uniform distribution from which the average (across species) annual intercept of p is drawn

sig.lp	sd of the normal distribution from which species-specific detection intercepts are drawn (centered on $\text{logit}(\text{mean.p})$, which are year-specific), on logit scale
mu.beta.lp	community mean of the coefficients of the covariate in probabilities of detection: the probability-scale mean of the normal distribution from which the species-specific coefficients are drawn.
sig.beta.lp	sd of the normal distribution from which species-specific detection slopes are drawn (centered on mu.beta.lp)
range.beta1.season	the range of the annual variation in the linear effect of season (i.e., of month 1-12 if nrep = 12) on the product of availability and detection
range.beta2.season	the range of the annual variation in the quadratic effect of season (i.e., of month 1-12 if nrep = 12) on the product of availability and detection
range.sd.site	sd of normal distribution to model logit-normal noise in p at the site level in the first and the last year of the simulation, with values for intermediate years interpolated linearly
range.sd.survey	sd of normal distribution to model logit-normal noise in p at the occasion level, in the first and the last year, with values for intermediate years interpolated linearly
show.plot	if TRUE, plots are produced. Set this to FALSE when running many simulations.

Value

A list with the values of the input arguments and the following additional elements:

Xpsi1	Site covariate for psi1, a nsite x 1 matrix
Xphi	Yearly-site covariate for phi, a nsite x nyear matrix
Xgamma	Yearly-site covariate for gamma, a nsite x nyear matrix
Xp	Observation covariate for p, a nsite x nrep x nyear array
beta0.lpsi	initial (logit-scale) occupancy intercept for each species in the community, a vector of length nspec
beta1.lpsi	initial (log-scale) occupancy slope on Xpsi1 for each species in the community, a vector of length nspec
psi	occupancy probability per site, year and species, a nsite x nyear x nspec array
mean.phi	mean persistence (across species) intercept for each interval, a vector of length (nyear - 1)
mean.gamma	mean colonisation (across species) intercept for each interval, a vector of length (nyear - 1)
eps.lphi	additive species effects in $\text{logit}(\text{phi})$ intercept, a vector of length nspec
eps.lgamma	additive species effects in $\text{logit}(\text{gamma})$ intercept, a vector of length nspec
beta0.lphi	logit-scale persistence intercepts for each species in community, a nspec x (nyear - 1) matrix

beta0.lgamma	logit scale colonisation intercepts for each species in the community, a nspec x (nyear - 1) matrix
beta1.lphi	slope of logit(phi) on Xphi for each species in the community, a vector of length nspec
beta1.lgamma	slope of logit(gamma) on Xgamma for each species in the community, a vector of length nspec
phi	probability of persistence for each site, yearly interval and species, a nsite x (nyear-1) x nspec array
gamma	probability of colonisation for each site, yearly interval and species, a nsite x (nyear-1) x nspec array
mean.p	mean detection (across species) intercept for each year, a vector of length nyear
eps.lp	additive species effects in logit(p) intercept, a vector of length nspec
beta0.lp	species- and site-specific intercepts in the linear predictor for p, a nspec x nyear matrix
beta1.lp	species specific slopes of logit(p) on Xp, a vector of length nspec
beta1	linear effect of the occasion number on detection probability, a vector of length nyear
beta2	quadratic effect of the occasion number on detection probability, a vector of length nyear
sd.site	standard deviation of the zero-mean normal distribution from which additional, site-level detection heterogeneity is simulated, a vector of length nyear
sd.survey	standard deviation of the zero-mean normal distribution from which additional, occasion-level detection heterogeneity is simulated,, a vector of length nyear
eps1	additive site random effects tht generate unstructured site-level detection heterogeneity, a vector of length nsite
eps2	additive occasion random effects tht generate unstructured site-level detection heterogeneity, a vector of length nrep
n.occ	Number of occupied sites, a nyear x nspec matrix
psi.fs	Finite-sample occupancy proportion, a nyear x nspec matrix
mean.psi	Average psi over sites, a nyear x nspec matrix
z.obs	Observed value of z matrix, a nsite x nyear x nspec array
n.occ.obs	Observed number of occupied sites, a nyear x nspec matrix
psi.obs	Observed occupancy (finite sample), a nyear x nspec matrix
nyear.pres	Number of years when species present, a vector of length nspec
nspec.pres	Number of species ever present, scalar
nyear.det	Number of years when species detected, a vector of length nspec
nspec.det	Number of species ever detected, scalar
z	True value of z matrix (ie, presence/absence), a nsite x nyear x nspec array
p	Probability of detection, a nsite x nrep x nyear x nspec array
y	Observed detection history, a nsite x nrep x nyear x nspec array of 0/1

Author(s)

Marc Kéry, 28 Nov 2016

References

Kéry & Royle (2017) *Applied Hierarchical Modeling in Ecology* Section 17.2

Examples

```
# Simulate a data set with the default arguments and look at the structure of the output:
tmp <- simDCM()
str(tmp)

# Default arguments, without plots
str(data <- simDCM(show.plot = FALSE))
# More examples:
str(data <- simDCM(nspec = 200)) # More species (looks great)
str(data <- simDCM(nspec = 1))  # A single species (ha, works !)
str(data <- simDCM(nsite = 267)) # More sites
str(data <- simDCM(nsite = 1))  # A single site
str(data <- simDCM(nrep = 10))  # More visits
str(data <- simDCM(nyear = 25)) # More years
str(data <- simDCM(nyear = 2))  # Just two years
# str(data <- simDCM(nyear = 1)) # A single year ... crashes

# No species heterogeneity in parameters of initial occupancy
str(data <- simDCM(sig.lpsi1 = 0, sig.beta.lpsi1 = 0))
# No species heterogeneity in parameters of persistence
str(data <- simDCM(sig.lphi = 0, sig.beta.lphi = 0))
# No species heterogeneity in parameters of colonisation
str(data <- simDCM(sig.lgamma = 0, sig.beta.lgamma = 0))
# No species heterogeneity in parameters of detection
str(data <- simDCM(sig.lp = 0, sig.beta.lp = 0))
# No annual variation in rates
str(data <- simDCM(range.mean.phi = c(0.8, 0.8), range.mean.gamma = c(0.3, 0.3),
  range.mean.p = c(0.6, 0.6)))

# Function arguments that lead to much structure (no zero arguments)
str(data <- simDCM(nspec = 200, nsite = 267, nrep = 3, nyear = 25,
  mean.psi1 = 0.4, sig.lpsi1 = 3, mu.beta.lpsi1 = 1, sig.beta.lpsi1 = 3,
  range.mean.phi = c(0.5, 1), sig.lphi = 3, mu.beta.lphi = 1,
  sig.beta.lphi = 3, range.mean.gamma = c(0, 0.5),
  sig.lgamma = 3, mu.beta.lgamma = -1, sig.beta.lgamma = 3,
  range.mean.p = c(0.1, 0.9), sig.lp = 3, mu.beta.lp = 1, sig.beta.lp = 3,
  range.beta1.season = c(-2, -0.5), range.beta2.season = c(0, 2),
  range.sd.site = c(0, 0), range.sd.survey = c(0, 0), show.plot = TRUE))

# Not every occurring species will be detected
set.seed(1)
str(data <- simDCM(nspec = 200, nsite = 20, nrep = 2, nyear = 10,
  mean.psi1 = 0.1, sig.lpsi1 = 5,
```

```

range.mean.phi = c(0.3, 0.3), sig.lphi = 5,
range.mean.gamma = c(0.1, 0.1), sig.lgamma = 5,
range.mean.p = c(0.1, 0.1), sig.lp = 5) )

# Pull out data from species 5
ysp5 <- data$y[, , 5]

# Pull out data from year 1
yyr1 <- data$y[, , 1, ]

```

simDynocc

Simulate data under a non-spatial dynamic occupancy model

Description

Function to simulate detection/nondetection data under a general dynamic site-occupancy model, including:

- * annual variation in the probabilities of patch persistence, colonization and detection is specified by the bounds of a uniform distribution.
- * one covariate is allowed to affect a parameter: a site covariate for ψ_1 , site-by-year covariates for ϕ and γ , and an observational covariate for p
- * Additional detection heterogeneity at the site-, the survey, or the site-by-survey level, with the possibility of a temporal trend in this heterogeneity over the years. E.g., an annual trend in detection heterogeneity at the site or the survey level is specified by the first and second value, which correspond to the heterogeneity in the first and the last year. Hence, $\text{range.sd.site} = c(0, 1)$ will result in a linear trend in the magnitude of site heterogeneity in detection from 0 in the first year to 1 in the last year.
- * Additional detection heterogeneity that varies over the season (= occasion) according to a quadratic effect of occasion number (to model phenology of an insect species for instance).
- * Simulation of data under a BACI (before-after-control-impact) design, where some event happens in a given year and *reduces* ϕ or γ by a stated percentage (only reductions, no increases allowed !)

Usage

```

simDynocc(nsite = 250, nrep = 3, nyear = 10, year.impact = 5,
mean.psi1 = 0.4, beta.Xpsi1 = 0,
range.phi = c(0.5, 1), impact.phi = 0, beta.Xphi = 0,
range.gamma = c(0, 0.5), impact.gamma = 0, beta.Xgamma = 0,
range.p = c(0.1, 0.9), beta.Xp = 0,
range.beta1.season = c(0, 0), range.beta2.season = c(0, 0),
range.sd.site = c(0, 0), range.sd.survey = c(0, 0),
range.sd.site.survey = c(0, 0),
show.plot = TRUE)

```

Arguments

nsite	Number of sites
nrep	Number of replicate surveys within a year (= season)
nyear	Number of years (or 'seasons')
year.impact	Year when some impact happens (for BACI design)
mean.psi1	average occupancy probability in first year
beta.Xpsi1	coefficients of environmental covariate in probability of initial occupancy.
range.phi	bounds of uniform distribution from which annual persistence is drawn
impact.phi	effect in percent on annual phi (must be zero or negative, e.g., impact.phi = -20 for a 20% reduction in phi)
beta.Xphi	coefficients of environmental covariate in probability of persistence.
range.gamma	bounds of uniform distribution from which annual colonisation is drawn
impact.gamma	effect in percent on annual gamma (must be zero or negative, e.g., impact.gamma = -20 for a 20% reduction in gamma)
beta.Xgamma	coefficients of environmental covariate in probability of colonization.
range.p	bounds of uniform distribution from which annual p is drawn
beta.Xp	coefficients of environmental covariate in probability of detection.
range.beta1.season	the range of the annual variation in the linear effect of season (i.e., of month 1-12) on the product of availability and detection linear and quadratic effect of season
range.beta2.season	the same for the quadratic effect of season
range.sd.site	sd of normal distribution to model logit-normal noise in p at the site level in the first and the last year of the simulation; if the two values in the range are the same, a constant value is assumed over time, while if they are different, a linear trend is assumed over time.
range.sd.survey	sd of normal distribution to model logit-normal noise in p <i>only</i> at the rep = 'survey' level, in the first and the last year; if they are different, a linear trend is assumed over time.
range.sd.site.survey	sd of normal distribution to model logit-normal noise in p at the site/year/rep = 'survey' level, in the first and the last year; if they are different, a linear trend is assumed over time.
show.plot	If TRUE, plots of results are displayed; set to FALSE if running many simulations.

Value

A list with the values of the arguments input and the following additional elements:

impact a 0/1 vector of length nyear - 1 indicating if an impact applies to the interval

phi.effect	additive effect of impact on persistence, a vector of length nyear - 1
gamma.effect	additive effect of impact on colonisation, a vector of length nyear - 1
beta1	linear effect of occasion on the product of availability and detection, a vector of length nyear
beta2	quadratic effect of occasion on the product of availability and detection, a vector of length nyear
mean.phi	mean persistence for each interval before application of any BACI effect, a vector of length nyear - 1
mean.gamma	mean colonisation for each interval before application of any BACI effect, a vector of length nyear - 1
mean.p	mean detection probability for each year, a vector of length nyear
psi	annual occupancy for each site, a nsite x nyear matrix
mean.psi	average occupancy over sites, a vector of length nyear
n.occ	number of occupied sites, a vector of length nyear
psi.fs	finite-sample occupancy proportion, a vector of length nyear
psi.app	apparent occupancy over sites, a vector of length nyear
z	true occurrence state, a nsite x nyear matrix of 0/1
phi	persistence, a nsite x nyear-1 matrix
gamma	colonisation, a nsite x nyear-1 matrix
p	detection probability, a nsite x nrep x nyear array
y	the observed detection history, a nsite x nrep x nyear array
Xpsi1	covariate affecting initial occupancy, a vector of length nsite
Xphi	covariate affecting persistence, a nsite x nyear matrix
Xgamma	covariate affecting colonisation, a nsite x nyear matrix
Xp	covariate affecting probability of detection, a nsite x nrep x nyear array
eps3	the array of site/rep/year random effects, a nsite x nrep x nyear array

Author(s)

Marc Kéry, 4 Dec 2014, modified 18-20 October 2016

References

Kéry & Royle (2017) *Applied Hierarchical Modeling in Ecology* Chapter 16

Examples

```
# Generate data with the default arguments and look at the structure:
tmp <- simDynocc()
str(tmp)

# no annual variation in the parameters
str(data <- simDynocc(nsite = 250, nrep = 3, nyear = 10, mean.psi1 = 0.6,
```



```

    range.phi = c(0.7, 0.7), range.gamma = c(0.3, 0.3), range.p = c(0.5, 0.5)))
# a fully time-dependent model (with p constant within each primary period)
str(data <- simDynocc(mean.psi1 = 0.6, range.phi = c(0.5, 0.8),
  range.gamma = c(0.1, 0.5), range.p = c(0.1, 0.9)) )
# a time-constant model with four different covariates affecting the four parameters
str(data <- simDynocc(mean.psi1 = 0.6, beta.Xpsi1 = 1,
  range.phi = c(0.6, 0.6), beta.Xphi = 2, range.gamma = c(0.3, 0.3),
  beta.Xgamma = 2, range.p = c(0.2, 0.2), beta.Xp = -2) )
# seasonal variation in detection probability
str(data <- simDynocc(nrep = 12, mean.psi1 = 0.6,
  range.phi = c(0.6, 0.6), range.gamma = c(0.3, 0.3),
  range.p = c(0.5, 0.5), range.beta1.season = c(-0.3, 0.4),
  range.beta2.season = c(0, -0.7)) )
# now both yearly variation and effects of all covariates (including season)
str( data <- simDynocc(mean.psi1 = 0.6, beta.Xpsi1 = 1,
  range.phi = c(0.6, 1), beta.Xphi = 2, range.gamma = c(0, 0.2),
  beta.Xgamma = 2, range.p = c(0.1, 0.9), beta.Xp = -2,
  range.beta1.season = c(-0.4, 0.5), range.beta2.season = c(0, -0.8)) )

# To add detection heterogeneity at the site level, you can do this:
str(data <- simDynocc(range.sd.site = c(3, 3)) ) # No time trend
str(data <- simDynocc(range.sd.site = c(1, 3)) ) # With time trend

# To add detection heterogeneity at the level of the survey, you can do this:
str(data <- simDynocc(range.sd.survey = c(3, 3)) ) # No time trend
str(data <- simDynocc(range.sd.survey = c(1, 3)) ) # With time trend

# To add detection heterogeneity at the level of the individual visit, you can do this:
str(data <- simDynocc(range.sd.site.survey = c(3, 3)) ) # No trend
str(data <- simDynocc(range.sd.site.survey = c(1, 3)) ) # With trend

# To simulate data under a BACI design, where an impact happens in year 10
str(data <- simDynocc(nsite = 250, nrep = 3, nyear = 20, year.impact = 10,
  impact.phi = -80, impact.gamma = -50) )

# And data where there is no detection error (i.e., with p = 1):
str( data <- simDynocc(range.p = c(1, 1)) )

```

simHDS

Simulate data under hierarchical distance sampling protocol (line or point)

Description

Function simulates hierarchical distance sampling (HDS) data under either a line or a point transect protocol.

Usage

```
simHDS(type=c("line", "point"), nsites = 100, mean.lambda = 2, beta.lam = 1,
       mean.sigma = 1, beta.sig = -0.5, B = 3, discard0 = TRUE, show.plot = TRUE)
```

Arguments

type	type of transect, "line" or "point".
nsites	Number of sites (spatial replication)
mean.lambda	the expected value of lambda when the habitat covariate = 0; the intercept of the log-linear regression for lambda is $\log(\text{mean.lambda})$.
beta.lam	the slope of the log-linear regression for lambda on a habitat covariate.
mean.sigma	the expected value of the scale parameter of the half-normal detection function when the wind speed = 0; the intercept of the log-linear regression for sigma is $\log(\text{mean.sigma})$.
beta.sig	the slope of log-linear regression of scale parameter of the half-normal detection function on wind speed
B	the strip half-width or circle radius
discard0	If TRUE, subset to sites at which individuals were captured. You may or may not want to do this depending on how the model is formulated so be careful.
show.plot	choose whether to show plots or not. Set to FALSE when using function in simulations.

Value

A list with the values of the arguments entered and the following additional elements:

data	simulated distance sampling data: a matrix with a row for each individual detected and 5 columns: site ID, status (1 if captured), x and y coordinates (NA for line transects), distance from the line or point; if <code>discard0 = FALSE</code> , sites with no detections will appear in the matrix with NAs in columns 2 to 5.
habitat	simulated habitat covariate
wind	simulated detection covariate
N	simulated number of individuals at each site
N.true	for point counts, the simulated number of individuals within the circle sampled

Author(s)

Kéry & Royle

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 8.5.1

Examples

```
# Simulate a data set with the default arguments and look at the structure of the output:
set.seed(123)
tmp <- simHDS()
str(tmp)
head(tmp$data)

tmp <- simHDS("point", discard0=FALSE)
str(tmp)
head(tmp$data, 10)
```

simHDSg

Simulate data under HDS protocol with groups

Description

Simulates hierarchical distance sampling (HDS) data for groups under either a line or a point transect protocol and using a half-normal detection function (Buckland et al. 2001).

Usage

```
simHDSg(type = c("line", "point"), nsites = 100, lambda.group = 0.75,
  alpha0 = 0, alpha1 = 0.5,
  beta0 = 1, beta1 = 0.5, B = 4, discard0 = TRUE, show.plot = TRUE)
```

Arguments

type	The type of distance transect, either "line" or "point".
nsites	Number of sites (spatial replication)
lambda.group	Poisson mean of group size
alpha0	intercept of log-linear model relating sigma of the half-normal detection function to group size
alpha1	slope of log-linear model relating sigma of the half-normal detection function to group size
beta0	intercept of log-linear model relating the Poisson mean of the number of groups per unit area to habitat
beta1	slope of log-linear model relating the Poisson mean of the number of groups per unit area to habitat
B	strip half width or the radius of the circle
discard0	whether to discard or keep the data from sites with nobody detected
show.plot	choose whether to show plots or not. Set to FALSE when using function in simulations.

Value

A list with the values of the arguments entered and the following additional elements:

data	simulated distance sampling data: a matrix with a row for each group detected and 6 columns: site ID, status (1 if captured), x and y coordinates (NA for line transects), distance from the line or point, group size; if <code>discard0 = FALSE</code> , sites with no detections will appear in the matrix with NAs in columns 2 to 6.
habitat	simulated habitat covariate
N	simulated number of groups at each site
N.true	for point counts, the simulated number of groups within the circle sampled
groupsize	group size for each of the groups observed

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 9.2.1

Buckland et al (2001) *Introduction to distance sampling: estimating abundance of biological populations*. Oxford University Press, Oxford, UK.

Examples

```
# Run with the default arguments and look at the structure of the output:
set.seed(123)
tmp <- simHDSg()
str(tmp)
head(tmp$data)

str(simHDSg(type = "line"))      # Defaults for line transect data
str(simHDSg(type = "point"))    # Defaults for point transect data
str(simHDSg(lambda.group = 5))  # Much larger groups
str(simHDSg(lambda.group = 5, alpha1 = 0)) # No effect of groups size on p
```

simHDSopen

Simulate open hierarchical distance sampling data

Description

Simulates distance sampling data from a multi-season (or multi-year) model, incorporating habitat and detection covariates, temporary emigration, and a trend in abundance or density.

Usage

```
simHDSopen(type=c("line", "point"), nsites = 100,
  mean.lam = 2, beta.lam = 0, mean.sig = 1, beta.sig = 0,
  B = 3, discard0 = TRUE, nreps = 2, phi = 0.7, nyears = 5, beta.trend = 0)
```

Arguments

type	the transect protocol, either "line" or "point" .
nsites	Number of sites (spatial replication)
mean.lam	intercept of log-linear regression of expected lambda on a habitat covariate
beta.lam	slope of log-linear regression of expected lambda on a habitat covariate
mean.sig	intercept of log-linear regression of scale parameter of half-normal detection function on wind speed
beta.sig	slope of log-linear regression of scale parameter of half-normal detection function on wind speed
B	strip half width, or maximum distance from the observer for point counts
discard0	Discard sites at which no individuals were captured. You may or may not want to do this depending on how the model is formulated so be careful.
nreps	the number of distance sampling surveys within a period of closure in a season (or year)
phi	the availability parameter
nyears	the number of seasons (typically years)
beta.trend	loglinear trend of annual population size or density

Value

A list with the values of the arguments entered and the following additional elements:

data	simulated distance sampling data: a list with a component for each year, each itself a list with a component for each replicate; this is a matrix with a row for each individual detected and 5 columns: site ID, status (1 if captured), x and y coordinates (NA for line transects), distance from the line or point; if <code>discard0 = FALSE</code> , sites with no detections will appear in the matrix with NAs in columns 2 to 5.
habitat	simulated habitat covariate, a vector of length <code>nsites</code>
wind	simulated detection covariate, a <code>nsites x nreps x nyears</code> array
M.true	simulated number of individuals, a <code>nsites x nyears</code> matrix
K	= <code>nreps</code>
Na	the number of individuals available for detection, a <code>nsites x nreps x nyears</code> array
Na.real	for point counts, the number of individuals available for detection within the circle sampled, a <code>nsites x nreps x nyears</code> array

Note

For "point" the realized density is $[(\text{area of circle}) / (\text{area of square})] * \text{lambda}$

Author(s)

Marc Kéry & Andy Royle

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 9.5.4.1

Examples

```
set.seed(123)
tmp <- simHDSopen() # Generate data with default parameters
str(tmp)
head(tmp$data[[1]][[1]])

tmp <- simHDSopen("point")
str(tmp)
head(tmp$data[[1]][[1]])

tmp <- simHDSopen(discard0=FALSE)
str(tmp)
head(tmp$data[[1]][[1]])
```

simHDStr

Simulate HDS time-removal or double-observer data

Description

A general function for simulating hierarchical distance sampling (HDS) data combined with a time-removal (with 3 removal periods) or double-observer protocol, either for a line or a point transect protocol and with method = "removal" or method = "double". Also produces plots of the output.

Usage

```
simHDStr(type = c("line", "point"), method=c("removal", "double"), nsites = 200,
  lambda.group = 1, alpha0 = 0, alpha1 = 0, beta0 = 1, beta1 = 0.5,
  p.avail = 0.75, K = 3, p.double = c(0.4, 0.6),
  B = 3, discard0 = FALSE, show.plot = TRUE)
```

Arguments

type	The type of distance transect, either "line" or "point".
method	Is the method time-removal ("removal") or double-observer ("double")
nsites	Number of sites (spatial replication)
lambda.group	Poisson mean of group size
alpha0	intercept of log-linear model relating sigma of half-normal detection function to group size
alpha1	slope of log-linear model relating sigma of half-normal detection function to group size
beta0	intercept of log-linear model relating the Poisson mean of the number of groups per unit area to habitat

beta1	slope of log-linear model relating the Poisson mean of the number of groups per unit area to habitat
p.avail	overall availability probability (phi in text)
K	number of removal periods (of equal length)
p.double	detection probability for first and second observer
B	strip half width or radius of the circle
discard0	whether to discard or keep the data from sites with nobody detected
show.plot	choose whether to show plots or not. Set to FALSE when running many simulations.

Value

A list with the values of the arguments and the following additional elements:

data	simulated distance sampling data: a matrix with a row for each group detected and 7 columns: site ID, status (1 if captured), x and y coordinates (NA for line transects), distance from the line or point, group size, the time interval of removal or capture history; if <code>discard0 = FALSE</code> , sites with no detections will appear in the matrix with NAs in columns 2 to 7.
habitat	simulated habitat covariate
M	simulated number of groups at each site
M.true	for point counts, the simulated number of groups within the circle sampled
params	a vector with the input arguments

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 9.3.2 (time removal) and 9.4.1 (double observer).

Examples

```
# Run with the default arguments and look at the structure of the output:
set.seed(123)
tmp <- simHDStr() # default: line, removal
str(tmp)
head(tmp$data)
tmp <- simHDStr("point", method="double")
str(tmp)
head(tmp$data)
```

simNmix

*Simulate data for binomial and multinomial mixture models***Description**

This very general function generates single-season count data under variants of the binomial N-mixture model of Royle (2004) and of the multinomial N-mixture model of Royle et al (2007).

Usage

```
simNmix(nsite = 267, nvisit = 3, mean.theta = 1, mean.lam = 2, mean.p = 0.6,
        area = FALSE, beta1.theta = 0, beta2.theta = 0, beta3.theta = 0,
        beta2.lam = 0, beta3.lam = 0, beta4.lam = 0, beta3.p = 0, beta5.p = 0,
        beta6.p = 0, beta.p.survey = 0, beta.p.N = 0, sigma.lam = 0, dispersion = 10,
        sigma.p.site = 0, sigma.p.visit = 0, sigma.p.survey = 0, sigma.p.ind = 0,
        Neg.Bin = FALSE, open.N = FALSE, show.plot = TRUE)
```

Arguments

nsite	number of sites
nvisit	number of visits per site
mean.theta	proportion of sites that can have non-zero abundance in principle: suitability model for zero-inflation
mean.lam	Expected abundance at the average value of all abundance covariates (and ignoring random site effects): abundance model
mean.p	Expected detection at the average value of all detection covariates (and ignoring all random effects): detection model
area	determines area of sites (A), defaults to A=1 (i.e., all identical), but you can supply a vector of site areas of length nsite instead.
beta1.theta	coefficient of site covariate 1 in suitability model
beta2.theta	coefficient of site covariate 2 in suitability model
beta3.theta	coefficient of site covariate 3 in suitability model
beta2.lam	coefficient of site covariate 2 in abundance model
beta3.lam	coefficient of site covariate 3 in abundance model
beta4.lam	coefficient of site covariate 4 in abundance model
beta3.p	coefficient of site covariate 3 in detection model
beta5.p	coefficient of site covariate 5 in detection model
beta6.p	coefficient of site covariate 6 in detection model
beta.p.survey	coefficient of survey ('observational') covariate on p
beta.p.N	coefficient of centered local population size ($\log(N+1)$) in detection model (i.e., coef. for density-dependent detection prob.)
sigma.lam	"Overdispersion SD" in linear predictor of abundance

dispersion	'size' or extra-Poisson dispersion of Negative binomial
sigma.p.site	"Overdispersion SD" in linear predictor of detection coming from random site effects
sigma.p.visit	"Overdispersion SD" in linear predictor of detection coming from random visit effects
sigma.p.survey	"Overdispersion SD" in linear predictor of detection coming from random site-by-survey effects
sigma.p.ind	"Overdispersion SD" in linear predictor of detection coming from random site-by-individual effects
Neg.Bin	if FALSE, any overdispersion in abundance is modelled by a Poisson log-normal; if TRUE, abundance overdispersion is modelled by adoption of a Negative binomial distribution for latent N
open.N	if TRUE, data are simulated under one specific form of an open population, where N in the first occasion is drawn from the specified mixture distribution and for all further occasions j, we have $N_{ij} \sim \text{Poisson}(N_{i(j-1)})$. With open.N = TRUE, we must have sigma.p.ind = 0, show.plot = FALSE and nvisit > 1.
show.plot	if TRUE, plots of the data will be displayed; set to FALSE if you are running many simulations.

Details

Data are simulated at the level of each individual and individual-specific detection heterogeneity can be included. As a side-effect, individual-specific detection histories are generated and hence, data are also be simulated under the corresponding multinomial N-mixture model.

Broadly, the function can generate data under this most general model:

'Suitability' (zero-inflation) $\sim \text{cov1} + \text{cov2} + \text{cov3}$

Abundance $\sim \text{offset} + \text{cov2} + \text{cov3} + \text{cov4} + \text{overdispersion}$

Detection $\sim \text{cov3} + \text{cov5} + \text{cov6} + \text{survey.covariate} + \log(N+1) + \text{eps.site} + \text{eps.visit} + \text{eps.survey} + \text{eps.individual}$

Overdispersion in abundance is modelled either as a Poisson-log-normal with a normal random site effect in lambda or with a Negative binomial with mean lambda and a 'size', or dispersion, parameter. Variable site areas can be specified to affect abundance as in an offset.

Abundance can be zero-inflated (this is the 'suitability' model). Note that the zero-inflation parameter is called theta here (in unmarked it is called psi). mean.phi is the probability that a site is suitable (i.e., 1 minus the expected proportion of sites with structural zero abundance).

Site covariate 2 can affect both suitability and abundance, while covariate 3 may affect all three levels. Hence, the function permits to simulate the case where a single site covariate affects different levels in the process (e.g., abundance and detection) in opposing directions (as for instance in Kéry, Auk, 2008)

Density-dependent detection can be modelled as a logistic-linear effect of local abundance (centered and $\log(x+1)$ transformed). Overdispersion in detection is modelled via normal random effects (the eps terms above) specific to sites, visits, surveys or individuals.

Effects of covariates and random-effects factors are modelled as additive on the link scale (log for abundance and logit for suitability and detection).

Data may be generated under one specific open-population model when argument 'open.N' is set to TRUE.

Value

A list with the arguments input and the following additional elements:

nobs	The total number of visits
site.cov	An nsite x 6 matrix of values for 6 site covariates
survey.cov	An nsite x nvisit matrix of values for a survey covariate
log.lam	Linear predictor of PLN abundance model including random effects, a vector of length nsites
s	Site suitability indicator, a vector of length nsites
N	Number of individuals at each site, a vector of length nsites
p	Probability of detection, an array with dimensions sites occupied x visits x max(N)
DH	Detection history (1/0), an array with dimensions sites occupied x visits x max(N)
N.open	Number of individuals at each site for open model, a sites x visits matrix
C	Summary of DH: number of individuals detected for each site and visit
eta.lam	Random site effects in lambda, a vector of length nsites, zero if Neg.Bin == TRUE
eta.p.site	Random site effects in p, a vector of length nsites
eta.p.visit	Random visit effects in p, a vector of length nvisits
eta.p.survey	Random survey (= site-by-survey) effects in p, a nsites x nvisits matrix
eta.p.ind	Random individual (= site-by-ind) effects in p (NOT site-ind-visit !), a nsites x max(N) matrix
odcN	Naive overdispersion measure (var/mean) for true abundance (N)
odcC	Naive overdispersion measure (var/mean) for observed counts (C)
Ntotal	Total abundance summed over all sites
summax	The sum of maximum counts over all sites

Author(s)

Marc Kéry, 2014-2015

References

- Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 6.5
- Royle (2004) N-mixture models for estimating population size from spatially replicated counts. *Biometrics* 60, 108-115
- Royle et al (2007) Hierarchical spatial models of abundance and occurrence from imperfect survey data. *Ecological Monographs*, 77, 465-481.
- Kéry (2008) Estimating abundance from bird counts: binomial mixture models uncover complex covariate relationships, *Auk* 125, 336-345

Examples

```
# Generate data with the default arguments and look at the structure:
tmp <- simNmix()
str(tmp)

str(data <- simNmix()) # Null data-generating model
str(data <- simNmix(mean.theta = 0.60)) # ZIP with 40% structural zeroes
str(data <- simNmix(sigma.lam = 1)) # Poisson-lognormal (PLN) mixture
str(data <- simNmix(Neg.Bin = TRUE)) # Negative-binomial mixture
str(data <- simNmix(mean.theta = 0.6, sigma.lam = 1)) # Zero-inflated PLN
str(data <- simNmix(mean.theta = 0.6, Neg.Bin = TRUE)) # Zero-infl. NegBin
str(data <- simNmix(mean.p = 1)) # Perfect detection (p = 1)
str(data <- simNmix(mean.theta = 0.6, mean.p = 1)) # ZIP with p = 1
str(data <- simNmix(sigma.lam = 1, mean.p = 1)) # PLN with p = 1
```

simOcc	<i>Simulate data for static occupancy models under wide range of conditions</i>
--------	---

Description

Function to simulate occupancy measurements replicated at M sites during J occasions. Population closure is assumed for each site. Expected occurrence may be affected by elevation (elev), forest cover (forest) and their interaction. Expected detection probability may be affected by elevation, wind speed (wind) and their interaction.

Usage

```
simOcc(M = 267, J = 3, mean.occupancy = 0.6, beta1 = -2, beta2 = 2, beta3 = 1,
       mean.detection = 0.3, time.effects = c(-1, 1),
       alpha1 = -1, alpha2 = -3, alpha3 = 0, sd.lp = 0.5,
       b = 2, show.plot = TRUE)
```

Arguments

M	Number of spatial replicates (sites)
J	Number of temporal replicates (occasions)
mean.occupancy	Mean occurrence at value 0 of occurrence covariates
beta1	Main effect of elevation on occurrence
beta2	Main effect of forest cover on occurrence
beta3	Interaction effect on occurrence of elevation and forest cover
mean.detection	Mean detection prob. at value 0 of detection covariates
time.effects	bounds (on logit scale) for uniform distribution from which time effects gamma will be drawn

alpha1	Main effect of elevation on detection probability
alpha2	Main effect of wind speed on detection probability
alpha3	Interaction effect on detection of elevation and wind speed
sd.lp	standard deviation of random site effects (on logit scale)
b	constant value of 'behavioural response' leading to 'trap-happiness' (if $b > 0$) or 'trap shyness' (if $b < 0$)
show.plot	if TRUE, plots of the data will be displayed; set to FALSE if you are running many simulations

Value

A list with the values of the input arguments and the following additional elements:

gamma	The time effects, a vector of length J
eps	Individual random effects, a vector of length M
elev	Elevation, a vector of length M
forest	Forest cover, a vector of length M
wind	wind speed, a $M \times J$ matrix
psi	Probability of occurrence, a vector of length M
z	Realised occurrence (0/1), a vector of length M
p	probability of capture, possibly with a behavioural effect, a $M \times J$ matrix
p0	probability of capture when not captured on previous occasion, a $M \times J$ matrix
p1	probability of capture when captured on previous occasion, a $M \times J$ matrix
y	simulated capture history, a $M \times J$ matrix
sumZ	True number of occupied sites
sumZ.obs	Number of sites observed to be occupied
psi.fs.true	True proportion of occupied sites in sample (sumZ/N)
psi.fs.obs	Proportion of sites observed to be occupied ($\text{sumZ.obs}/N$)

Author(s)

Marc Kéry

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 10.8

Examples

```

# Generate data with the default arguments and look at the structure:
tmp <- simOcc()
str(tmp)

# Simplest possible occupancy model, with constant occupancy and detection
str(simOcc(mean.occ=0.6, beta1=0, beta2=0, beta3=0, mean.det=0.3, time.effects=c(0, 0),
  alpha1=0, alpha2=0, alpha3=0, sd.lp=0, b=0))
# psi = 1 (i.e., species occurs at every site)
str(simOcc(mean.occ=1))

# p = 1 (i.e., species is always detected when it occurs)
str(simOcc(mean.det=1))

# Other potentially interesting settings include these:
str(simOcc(J = 2))           # Only 2 surveys
str(simOcc(M = 1, J = 100)) # No spatial replicates, but 100 measurements
str(simOcc(beta3 = 1))      # Including interaction elev-wind on p
str(simOcc(mean.occ = 0.96)) # A really common species
str(simOcc(mean.occ = 0.05)) # A really rare species
str(simOcc(mean.det = 0.96)) # A really easy species
str(simOcc(mean.det = 0.05)) # A really hard species
str(simOcc(mean.det = 0))    # The dreaded invisible species
str(simOcc(alpha1=-2, beta1=2)) # Opposing effects of elev on psi and p
str(simOcc(J = 10, time.effects = c(-5, 5))) # Huge time effects on p
str(simOcc(sd.lp = 10))     # Huge (random) site effects on p
str(simOcc(J = 10, b = 0))   # No behavioural response in p
str(simOcc(J = 10, b = 2))   # Trap happiness
str(simOcc(J = 10, b = -2))  # Trap shyness

```

simOccttd

Simulate time-to-detection occupancy data

Description

Function simulates time-to-detection occupancy design data under model of Garrard et al. (2008), also see Bornand et al. (2014)

Usage

```

simOccttd(M = 250, mean.psi = 0.4, mean.lambda = 0.3,
  beta1 = 1, alpha1 = -1, Tmax = 10, show.plot = TRUE)

```

Arguments

M	Number of sites
mean.psi	intercept of occupancy probability
mean.lambda	intercept of Poisson rate parameter

beta1	slope of continuous covariate B on logit(psi)
alpha1	slope of continuous covariate A on log(lambda)
Tmax	maximum search time (in arbitrary units, which are same as response), response will be censored at Tmax
show.plot	choose whether to show plots or not. Set to FALSE when running many simulations.

Value

A list with the values of the arguments input and the following additional elements:

covA	Simulated values of covariate A, a vector of length M
covB	Simulated values of covariate B, a vector of length M
psi	Probability of occurrence at each site, a vector of length M
z	Realised occurrence at each site, a 0/1 vector of length M
ttd.temp	Uncensored simulated time-to-detection at each site, a vector of length M
ttd	Censored simulated time-to-detection at each site, a vector of length M
d	Censoring indicator, a 0/1 vector of length M
sum.z	Total number of sites occupied
n.obs	Total number of sites where the species was observed

Author(s)

Marc Kéry

References

- Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 10.12.1
- Garrard, G.E., Bekessy, S.A., McCarthy, M.A., & Wintle, B.A. (2008) When have we looked hard enough? A novel method for setting minimum survey effort protocols for flora surveys. *Austral Ecology*, 33, 986-998.
- Bornand, C.N., Kéry, M., Bueche, L., & Fischer, M. (2014) Hide and seek in vegetation: time-to-detection is an efficient design for estimating detectability and occurrence. *Methods in Ecology and Evolution*, 5, 433-442.

Examples

```
# Generate data with the default arguments and look at the structure:
tmp <- simOccttd()
str(tmp)
```

simpleNmix

*Simulate N-mixture data under a time-for-space substitution design***Description**

A simple function to simulate data under binomial N-mixture model where you have a single site that is surveyed over 'nyear' primary sampling periods ('seasons', 'years'), within each of which there are 'nrep' secondary samples.

Usage

```
simpleNmix(nyear = 12, nrep = 4, beta0 = 2, beta1 = 0.1,
          alpha0 = 0.5, alpha1 = -0.1, alpha2 = 1, show.plot = TRUE)
```

Arguments

nyear	Number of primary sampling periods.
nrep	Number of secondary samples within each primary period.
beta0	the intercept of a log-linear model of expected abundance (λ).
beta1	the Time coefficient of a log-linear model for λ .
alpha0	the intercept of a logit-linear model for detection (p).
alpha1	the Time coefficient of a logit-linear model for detection (p).
alpha2	the coefficient of a survey-specific covariate such as temperature ($temp$).
show.plot	choose whether to show plots or not. Set to FALSE when using function in simulations.

Value

A list with the values of the arguments input and the following additional elements:

N	The realised number of individuals at each primary season, a vector of length nyear
C	The number of individuals counted at each survey, a nyear \times nrep matrix.
Time	The Time covariate, a vector of length nyear.
temp	The temperature covariate, a nyear \times nrep matrix.
p	The probability of detection, a nyear \times nrep matrix.

Author(s)

Kéry & Royle

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 6.12

Examples

```
# Simulate a data set with the default arguments and look at the structure of the output:
tmp <- simpleNmix()
str(tmp)
```

spatial.exp

Generate random field with negative exponential correlation

Description

Function creates Gaussian random field with negative exponential correlation and visualizes correlation and random field

Usage

```
spatial.exp(variance = 1, theta = 1, size = 50, show.plot = TRUE)
```

Arguments

variance	variance of field
theta	parameter governing spatial correlation ($=1/\phi$) ("large theta means high correlation") Note that RMexp is specified in terms of $\phi = 1/\theta$
size	Number of pixels in either direction
show.plot	if TRUE, plots of the data will be displayed; set to FALSE if you are running many simulations or use inside other functions.

Value

A list with the following components:

variance	the value of the variance argument
theta	the value of the theta argument
field	the random field variable, a size x size matrix
grid	the grid corresponding to field, equivalent to <code>expand.grid(1:size, 1:size)</code>

Author(s)

Kéry & Royle

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Chapter 20

Examples

```
# Generate data with the default arguments and look at the structure:
tmp <- spatial.exp()
str(tmp)
```

`spline.prep`*Prepare input for BUGS model when fitting a spline for a covariate*

Description

Function chooses knots and creates design matrices for fixed and random-effects parts of a spline model for a chosen covariate. Based on code by Crainiceanu et al. (2005) and Zuur et al. (2012). Allows you to choose number of knots or else uses it by the rule given in Crainiceanu et al. (2005). Prepares fixed part of covariate as a quadratic polynomial.

Usage

```
spline.prep(cov, nknot = NA)
```

Arguments

<code>cov</code>	the covariate, a numeric vector
<code>nknot</code>	optional, number of knots

Value

A list with the following elements:

<code>cov</code>	The input covariate
<code>knots</code>	The values of the knots
<code>X</code>	The fixed-effects design matrix
<code>Z</code>	The random-effects design matrix

Author(s)

Kéry & Royle

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 10.14

Crainiceanu, C.M., Ruppert, D., & Wand, M.P. (2005) Bayesian analysis for penalized spline regression using WinBUGS. *Journal of Statistical Software*, 14.

Zuur, A.F., Saveliev, A.A., Ieno, E.N. (2012) *Zero-inflated Models and Generalized Linear Mixed Models with R*. Highlands Statistics

Examples

```
## TO DO
```

SwissSquirrels

Data for red squirrels in Switzerland

Description

A file with detection/nondetection data for the red squirrel in 265 1 km² survey quadrats in Switzerland for 2007, together with covariates. See Examples for code to load the data.

Format

The file `SwissSquirrels.txt` is a tab-delimited text file with 265 rows and the following columns:

1. `spec.name` : the species name.
2. `coordx` : the x coordinate of the centre of the quadrat; the coordinate reference system intentionally not specified.
3. `coordy` : the y coordinate of the centre of the quadrat.
4. `ele` : the mean elevation of the quadrat, m.
5. `route` : the length of the route walked in the quadrat, km.
6. `forest` : percentage forest cover.
7. `det071`, `det072`, `det073` : 1/0 detection data for 3 survey occasions in 2007.
8. `date071`, `date072`, `date073` : Julian date for the 3 survey occasions (1 = 1st April).
9. `dur071`, `dur072`, `dur073` : duration of the 3 survey occasions (mins).

Source

Swiss Ornithological Institute

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 10.9

Examples

```
# To read in the text file:
fn <- file.path(find.package("AHMbook"), "extdata", "SwissSquirrels.txt")
data <- read.table(fn, header = TRUE)
str(data)
```

Description

The Swiss breeding bird survey ("Monitoring Häufige Brutvögel" MHB) has monitored the populations of 150 common species since 1999. The MHB sample consists of 267 1-km squares that are laid out as a grid across Switzerland. Fieldwork is conducted by about 200 skilled birdwatchers, most of them volunteers. Avian populations are monitored using a simplified territory mapping protocol, where each square is surveyed up to three times during the breeding season (only twice above the tree line). Surveys are conducted along a transect that does not change over the years.

The list `SwissTits` has the data for six species of tits from 2004 to 2013. There are some missing values: see `Details`.

Usage

```
data("SwissTits")
```

Format

`SwissTits` is a list with 5 elements:

species a data frame with a row for each species and the following columns:

1. `specid` : a numeric species ID based on phylogeny.
2. `latname` : a 6-letter abbreviation of the Latin name.
3. `name` : the English name.

sites a data frame with a row for each 1x1 km quadrat and the following columns:

1. `siteID` : an alphanumeric site identifier.
2. `coordx` : the x coordinate of the centre of the quadrat; the coordinate reference system intentionally not specified.
3. `coordy` : the y coordinate of the centre of the quadrat.
4. `AQ` : an identifier for the 10km x 10km "Atlas Quadrat" within which the site falls.
5. `AQ.coordx` : the x coordinate of the centre of the AQ.
6. `AQ.coordy` : the y coordinate of the centre of the AQ.
7. `elev` : the mean elevation of the quadrat, m.
8. `r.length` : the length of the route walked in the quadrat, km.
9. `forest` : percentage forest cover.

counts a sites x replicates x years x species array of counts

date a sites x replicates x years array with Julian dates of the surveys, 1 April = 1

dur a sites x replicates x years array with the duration of each survey, mins

Details

Missing values in the date array indicate that the corresponding survey was not carried out.

On 26 occasions when surveys were carried out, the duration was not recorded, resulting in additional NAs in the dur array.

A new method for recording breeding territories was introduced in 2004, but the old protocol was in use at some sites until 2013. Surveys with the old protocol have the counts shown as NA in the count array.

Note

Sections 6.9.1 and 6.13.1 of the book have code to read in data from a CSV file, "SwissTits_mhb_2004_2013.csv". The `SwissTits` list has all the same data in a more compact format. See Examples for ways to generate the objects used in the book from the list.

Source

Swiss Ornithological Institute

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Sections 6.9.1 and 6.13.1.

Examples

```
data(SwissTits)
str(SwissTits)

# Section 6.9.1, foot of p.257 and top of p.258:
# -----
y0 <- SwissTits$counts[, , '2013', 'Great tit']
( NA.sites <- which(rowSums(is.na(y0)) == 3) ) # Unsurveyed sites
y <- y0[-NA.sites, ] # Drop them from the count data
tits <- SwissTits$sites[-NA.sites, ] # Also drop from the site covariates
str(y)
# Get date and duration data for 2013, without the NA.sites rows:
date <- SwissTits$date[-NA.sites, , '2013']
dur <- SwissTits$dur[-NA.sites, , '2013']

# Section 6.13.1, p.303
# -----
# Get the count data for 2013 (all species)
y0 <- SwissTits$count[, , '2013', ]
str(y0)
# We keep the sites with count data, remove those with 3 NAs
# See which sites have counts in 2013 for (say) Great tits:
keep <- which(rowSums(is.na(y0[, , "Great tit"])) != 3)
length(keep)
y <- y0[keep, , ]
# Get the covariate data for the 'keep' sites
elev <- SwissTits$sites$ele[keep]
```

```

route <- SwissTits$sites$rlength[keep]
forest <- SwissTits$sites$forest[keep]
date <- SwissTits$date[keep, , '2013'] # Survey date
dur <- SwissTits$dur[keep, , '2013'] # Survey duration

# Degrade counts to detection/nondetection data
y3DRN <- y
y3DRN[y3DRN > 1] <- 1
str(y3DRN)
# Final detail...
( spec.names <- paste0(SwissTits$species$name, "s") )

```

ttdPeregrine

Time-to-detection data for peregrines from the French Jura mountains

Description

Between 7 and 9 March 2015, 38 breeding cliffs were visited up to 3 times. Observation duration varied from 3 to 95 minutes. The time to detection was recorded for each bird seen. If no birds were seen, the time was entered as NA.

Usage

```
data("ttdPeregrine")
```

Format

A data frame with 70 rows and the following columns:

Date a factor with 3 levels giving the date

DayNumber the number of the day: 1, 2, or 3

SiteNumber identification number of the site

Start.hour, Start.minute the time of starting the search

End.hour, End.minute the time of ending the search

MinOfDay the time of the start of the search, minutes after 06:00

Tmax the duration of the search, minutes

ttd the time to detection, minutes; NA if no birds were seen during the search

sex the sex of the birds seen; NA if no birds were seen

Source

Data provided with the AHM book at <https://www.mbr-pwrc.usgs.gov/pubanalysis/keryroylebook/>

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 10.12

Examples

```
# Using the built-in data set instead of the TXT file for the example in Section
# 10.12 of the book is easy, as the format is the same as the file.
?ttdPeregrine # check the description of the data
data(ttdPeregrine)
# Instead of data <- read.table("ttdPeregrine.txt", header = TRUE, sep = "\t") do:
data <- ttdPeregrine

# Then continue with the rest of the analysis on p.618
```

wagtail

Data for Dutch wagtails

Description

The Dutch Centre for Field Ornithology Sovon monitored grassland birds in Flevoland between April and mid-July 2011. 235 points were surveyed on up to 4 occasions; observations were divided into distance classes 50m wide, and the number of observations in each class recorded. These data are for yellow wagtails.

Usage

```
data("wagtail")
```

Format

wagtail is a list with 8 elements:

potato for each point, the percentage of the area which are potato fields.

grass for each point, the percentage of the area under permanent grassland.

lscale for each point, an index of whether the landscape is open (0) or closed (100).

date a points x occasions matrix, the Julian date of the survey.

hour a points x occasions matrix, the hour of the survey.

breaks the boundaries between the distance classes; birds more than 300m from the point were not included.

Y a matrix of counts, with a row for each site; columns 1 to 6 give the counts in the distance classes for the 1st survey occasion, columns 7 to 12 for the 2nd occasion, and so on (this is the format required for unmarkedFrameGDS).

rep a points x occasions character matrix with the occasion number (this is used as a categorical variable in the analysis).

Source

Data provided with the AHM book at <https://www.mbr-pwrc.usgs.gov/pubanalysis/keryroylebook/>

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 9.5.3

Examples

```
data(wagtail)
str(wagtail)
```

wigglyOcc	<i>Simulate static occupancy data</i>
-----------	---------------------------------------

Description

Function to generate a static occupancy data set with really wiggly covariate relationships in occupancy and detection probability

Usage

```
wigglyOcc(seed = 1, show.plot = TRUE)
```

Arguments

seed	Seed for random number generator
show.plot	choose whether to show plots or not. Set to FALSE when using function in simulations.

Value

A list with the following elements:

M	Number of sites
J	Number of replicate surveys
Xsite	Simulated site covariate, a vector of length M
Xsurvey	Simulated survey covariate, a M x J matrix
psi	Occupancy probability, a vector of length M
z	Realised occupancy, a 0/1 vector of length M
p	Detection probability, a M x J matrix
y	detection history, a M x J matrix of 0/1

Author(s)

Kéry & Royle

References

Kéry & Royle (2016) *Applied Hierarchical Modeling in Ecology* Section 10.14

Examples

```
# Generate data with the default arguments and look at the structure:  
tmp <- wigglyOcc()  
str(tmp)
```


Index

*Topic **datasets**

- dragonflies, 7
- Finnmark, 9
- MHB2014, 15
- SwissSquirrels, 58
- SwissTits, 59
- ttdPeregrine, 61
- wagtail, 62

*Topic **package**

- AHMbook-package, 2

AHMbook (AHMbook-package), 2

AHMbook-package, 2

bigCrossCorr, 5, 5

body (dragonflies), 7

color (dragonflies), 7

crosscorr, 5

crPiFun, 3

crPiFun (piFuns), 16

crPiFun.Mb, 3

damage (dragonflies), 7

data.fn, 3, 6

dragonflies, 4, 7

e2dist, 5, 8

Finnmark, 4, 9

fitstats, 3, 10

fitstats2, 3

fitstats2 (fitstats), 10

image.scale (image_scale), 11

image_scale, 5, 11

instRemPiFun, 3

instRemPiFun (piFuns), 16

issj.sim, 4, 12

jay, 4

map.Nmix.resi, 3, 14

MHB2014, 4, 15

MhPiFun, 3

MhPiFun (piFuns), 16

mites (dragonflies), 7

piFuns, 16, 16

playRN, 3, 18

plot.Nmix.resi (plot_Nmix_resi), 19

plot_Nmix_resi, 3, 19

pop (dragonflies), 7

ppc.plot, 3, 20

sex (dragonflies), 7

sim.fn, 3, 21

sim.ldata, 3, 22

sim.pdata, 3, 23

sim.spatialDS, 4, 24

sim.spatialHDS, 4, 26

sim30cc, 4, 28

simComm, 4, 30

simDCM, 4, 33

simDynocc, 4, 38

simHDS, 3, 41

simHDSg, 4, 43

simHDSopen, 4, 44

simHDSr, 4, 46

simNmix, 3, 48

simOcc, 4, 51

simOccttd, 4, 53

simpleNmix, 3, 55

spatial.exp, 4, 56

spline.prep, 4, 57

SwissSquirrels, 4, 58

SwissTits, 4, 59

ttdPeregrine, 4, 61

wagtail, 4, 62

wigglyOcc, 4, 63

wing (dragonflies), 7