Package ‘hSDM’

May 13, 2019

Type Package

Title Hierarchical Bayesian Species Distribution Models

Version 1.4.1

Date 2019-05-12

Imports coda

Suggests knitr, raster, sp, rmarkdown, bookdown

Maintainer Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>

Description User-friendly and fast set of functions for estimating parameters of hierarchical Bayesian species distribution models (Latimer et al. 2006 <doi:10.1890/04-0609>). Such models allow interpreting the observations (occurrence and abundance of a species) as a result of several hierarchical processes including ecological processes (habitat suitability, spatial dependence and anthropogenic disturbance) and observation processes (species detectability). Hierarchical species distribution models are essential for accurately characterizing the environmental response of species, predicting their probability of occurrence, and assessing uncertainty in the model results.

License GPL-3 | file LICENSE

URL https://ecology.ghislain.fr/hSDM,
https://github.com/ghislainv/hSDM

BugReports https://github.com/ghislainv/hSDM/issues

LazyLoad yes

Encoding UTF-8

RoxygenNote 6.1.1

VignetteBuilder knitr

NeedsCompilation yes

Author Ghislain Vieilledent [aut, cre]
(https://orcid.org/0000-0002-1685-4997),
Matthieu Autier [ctb],
Alan E. Gelfand [ctb],
Jérôme Guélat [ctb],
Marc Kéry [ctb].
hSDM-package

Andrew M. Latimer [ctb],
Cory Merow [ctb],
Frédéric Mortier [ctb],
John A. Silander Jr. [ctb],
Adam M. Wilson [ctb],
Shanshan Wu [ctb],
CIRAD [cph, fnd]

Repository CRAN

Date/Publication 2019-05-12 23:30:09 UTC

R topics documented:

hSDM-package .................................................. 2
altitude ......................................................... 3
cfr.env ......................................................... 4
data.Kery2010 ................................................... 5
datacells.Latimer2006 ......................................... 6
frogs ............................................................. 7
hSDM.binomial .................................................. 8
hSDM.binomial.iCAR ........................................... 11
hSDM.Nmixture ................................................ 18
hSDM.Nmixture.iCAR ........................................... 23
hSDM.Nmixture.K .............................................. 32
hSDM.poisson ................................................... 37
hSDM.poisson.iCAR .......................................... 40
hSDM.siteocc .................................................. 47
hSDM.siteocc.iCAR ........................................... 51
hSDM.ZIB ...................................................... 60
hSDM.ZIB.iCAR ................................................ 64
hSDM.ZIB.iCAR.alteration .................................... 71
hSDM.ZIP ........................................................ 79
hSDM.ZIP.iCAR ................................................. 83
hSDM.ZIP.iCAR.alteration .................................... 90
logit ........................................................... 98
neighbors.Latimer2006 ....................................... 99
punc10 ......................................................... 100

Index 101

hSDM-package    hierarchical Bayesian species distribution models
**Description**

hSDM is an R package for estimating parameters of hierarchical Bayesian species distribution models. Such models allow interpreting the observations (occurrence and abundance of a species) as a result of several hierarchical processes including ecological processes (habitat suitability, spatial dependence and anthropogenic disturbance) and observation processes (species detectability). Hierarchical species distribution models are essential for accurately characterizing the environmental response of species, predicting their probability of occurrence, and assessing uncertainty in the model results.

**Details**

<table>
<thead>
<tr>
<th>Package</th>
<th>hSDM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Package</td>
</tr>
<tr>
<td>Version</td>
<td>1.3</td>
</tr>
<tr>
<td>Date</td>
<td>2013-10-15</td>
</tr>
<tr>
<td>License</td>
<td>GPL-3</td>
</tr>
<tr>
<td>LazyLoad</td>
<td>yes</td>
</tr>
</tbody>
</table>

**Author(s)**

Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>
Cory Merow <cory.merow@gmail.com>
Jérôme Guélat <jerome.guelat@vogelwarte.ch>
Andrew M. Latimer <amlatimer@ucdavis.edu>
Marc Kéry <marc.kery@vogelwarte.ch>
Alan E. Gelfand <alan@stat.duke.edu>
Adam M. Wilson <adam.wilson@yale.edu>
Frédéric Mortier <frederic.mortier@cirad.fr>
John A. Silander Jr. <john.silander@uconn.edu>

---

**altitude**

**Virtual altitudinal data**

**Description**

Data frame with virtual altitudinal data. The data frame is used in the examples of the hSDM package vignette to derive an altitude raster determining species habitat suitability.
Usage

altitude

Format

altitude is a data frame with 2500 observations (50 x 50 cells) and 3 variables:

x coordinates of the center of the cell on the x axis
y coordinates of the center of the cell on the y axis
altitude altitude (m)

<table>
<thead>
<tr>
<th>cfr.env</th>
<th>Environmental data for South Africa’s Cap Floristic Region</th>
</tr>
</thead>
</table>

Description

Data include environmental variables for 36909 one minute by one minute grid cells on the whole South Africa’s Cap Floristic Region.

Usage

cfr.env

Format

cfr.env is a data frame with 36909 observations (cells) on the following six environmental variables.

lon longitude
lat latitude
min07 minimum temperature of the coldest month (July)
smdwin winter soil moisture days
fert3 moderately high fertility (percent of grid cell)
ph1 acidic soil (percent of grid cell)
text1 fine soil texture (percent of grid cell)
text2 moderately fine soil texture (percent of grid cell)

Source

Cory Merow’s personal data

References

Count data for the Willow tit (from Kéry and Royle 2010)

Description

Usage
data.Kery2010

Format
data.Kery2010 is a data frame with 264 observations (1 km^2 quadrats) and the following 10 variables.

- coordx  quadrat x coordinate
- coordy  quadrat y coordinate
- elevation mean quadrat elevation (m)
- forest  quadrat forest cover (in %)
- countQ  count for survey 1
- countR  count for survey 2
- countS  count for survey 3
- juldateQ Julian date of survey 1
- juldateR Julian date of survey 2
- juldateS Julian date of survey 3

Source

References
**Description**

Data come from a small region including 476 one minute by one minute grid cells. This region is a small corner of South Africa’s Cape Floristic Region, and includes very high plant species diversity and a World Biosphere Reserve. The data frame can be used as an example for several functions in the hSDM package.

**Usage**

datacells.Latimer2006

**Format**

datacells.Latimer2006 is a data frame with 476 observations (cells) on the following 9 variables.

- y: the number of times the species was observed to be present in each cell
- n: the number of visits or sample locations in each cell (which can be zero)
- rough: elevational range or "roughness"
- julmint: July minimum temperature
- pptcv: interannual variation in precipitation
- smdsum: summer soil moisture days
- evi: enhanced vegetation or "greenness" index
- ph1: percent acidic soil
- num: number of neighbors of each cell, this is a sparse representation of the adjacency matrix for the subregion.

**Source**

Latimer et al. (2006) *Ecological Applications*, Appendix B

**References**

Description
Counts of the number of frogs in ponds of the Canton Aargau, Switzerland.

Usage
data(frogs)

Format
A data frame with 481 observations on the following 10 variables.

- count1 number of counted frogs during the first visit
- count2 number of counted frogs during the second visit
- elevation elevation, meters above sea level
- year year
- fish presence of fish (1 = present, 0 = absent)
- waterarea area of the water body in square meters
- vegetation indicator of vegetation (1 = vegetation present, 0 = no vegetation present)
- pondid name of the pond, corresponds to observation id
- x x coordinate
- y y coordinate

Details
The amphibian monitoring program started in 1999 and is mainly aimed at surveying population trends of endangered amphibian species. Every year, about 30 water bodies in two or three randomly selected priority areas (out of ten priority areas of high amphibian diversity) are surveyed. Additionally, a random selection of water bodies that potentially are suitable for one of the endangered amphibian species but that do not belong to the priority areas were surveyed. Each water body is surveyed by single trained volunteer during two nocturnal visits per year. Volunteers recorded anurans by walking along the water’s edge with precise rules for the duration of a survey taking account of the size of the surveyed water body and noting visual encounters and calls. As fare as possible, encountered individuals of the Pelophylax-complex were identified as Marsh Frog (Pelophylax ridibundus), Pool Frog (P. lessonae) or hybrids (P. esculentus) based on morphological characteristics or based on their calls. In the given data set, however, these three taxa are lumped together.

Source
The data is provided by Isabelle Floess, Landschaft und Gewaesser, Kanton Aargau.
References


Examples

data(frogs)

---

hSDM.binomial  Binomial logistic regression model

Description

The hSDM.binomial function performs a Binomial logistic regression in a Bayesian framework. The function calls a Gibbs sampler written in C code which uses an adaptive Metropolis algorithm to estimate the conditional posterior distribution of model’s parameters.

Usage

hSDM.binomial(presences, trials, suitability, data, suitability.pred = NULL, burnin = 5000, mcmc = 10000, thin = 10, beta.start, mubeta = 0, Vbeta = 1e+06, seed = 1234, verbose = 1, save.p = 0)

Arguments

- **presences**: A vector indicating the number of successes (or presences) for each observation.
- **trials**: A vector indicating the number of trials for each observation. \( t_n \) should be superior or equal to \( y_n \), the number of successes for observation \( n \). If \( t_n = 0 \), then \( y_n = 0 \).
- **suitability**: A one-sided formula of the form ’~x1+...+xp’ with \( p \) terms specifying the explicative variables for the suitability process of the model.
- **data**: A data frame containing the model’s explicative variables.
- **suitability.pred**: An optional data frame in which to look for variables with which to predict. If NULL, the observations are used.
- **burnin**: The number of burnin iterations for the sampler.
- **mcmc**: The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to burnin+mcmc. burnin+mcmc must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.
- **thin**: The thinning interval used in the simulation. The number of mcmc iterations must be divisible by this value.
**Details**

We model an ecological process where the presence or absence of the species is explained by habitat suitability.

**Ecological process:**

\[
y_i \sim \text{Binomial}(\theta_i, t_i)
\]

\[
\text{logit}(\theta_i) = X_i \beta
\]

**Value**

- **mcmc**: An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance \( D \), with \( D = -2 \log(\prod_i P(y_i|\beta, t_i)) \), is also provided.
- **theta.pred**: If save.p is set to 0 (default), theta.pred is the predictive posterior mean of the probability associated to the suitability process for each prediction. If save.p is set to 1, theta.pred is an mcmc object with sampled values of the probability associated to the suitability process for each prediction.
- **theta.latent**: Predictive posterior mean of the probability associated to the suitability process for each observation.

**Author(s)**

Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>
References


See Also

plot.mcmc, summary.mcmc

Examples

```r
## Not run:

#-----------------------------
# hSDM.binomial()
# Example with simulated data
#-----------------------------

#-----------------------------
## Load libraries
library(hSDM)

#-----------------------------
## Data simulation

## Number of sites
nsite <- 200

## Set seed for repeatability
seed <- 1234

## Number of visits associated to each site
set.seed(seed)
visits <- rpois(nsite,3)
visits[visits==0] <- 1

## Ecological process (suitability)
set.seed(seed)
x1 <- rnorm(nsite,0,1)
set.seed(2*seed)
x2 <- rnorm(nsite,0,1)
X <- cbind(rep(1,nsite),x1,x2)
beta.target <- c(-1,1,-1)
logit.theta <- X %*% beta.target
theta <- inv.logit(logit.theta)
set.seed(seed)
Y <- rbinom(nsite,visits,theta)
```
# Data-sets
data.obs <- data.frame(Y, visits, x1, x2)

# Site-occupancy model
mod.hSDM.binomial <- hSDM.binomial(presences=data.obs$Y, 
trials=data.obs$visits, 
suitability=-x1+x2, 
data=data.obs, 
suitability.pred=NULL, 
burnin=1000, mcmc=1000, thin=1, 
beta.start=0, 
mubeta=0, Vbeta=1.0E6, 
seed=1234, verbose=1, 
save.p=0)

# Parameter estimates
summary(mod.hSDM.binomial$mcmc)
pdf(file="Posterior_hSDM.binomial.pdf")
plot(mod.hSDM.binomial$mcmc)
dev.off()

# glm resolution to compare
mod.glm <- glm(cbind(Y, visits-Y)-x1+x2, family="binomial", data=data.obs) 
summary(mod.glm)

# Predictions
summary(mod.hSDM.binomial$theta.latent) 
summary(mod.hSDM.binomial$theta.pred) 
pdf(file="Pred-Init.pdf")
plot(theta, mod.hSDM.binomial$theta.pred) 
abline(a=0,b=1,col="red")
dev.off()

# End (Not run)

---

**hSDM.binomial.iCAR**  
Binomial logistic regression model with CAR process

**Description**

The `hSDM.binomial.iCAR` function performs a Binomial logistic regression model in a hierarchical Bayesian framework. The suitability process includes a spatial correlation process. The spatial correlation is modelled using an intrinsic CAR model. The `hSDM.binomial.iCAR` function calls
a Gibbs sampler written in C code which uses an adaptive Metropolis algorithm to estimate the conditional posterior distribution of hierarchical model’s parameters.

Usage

```r
hsdm.binomial.iCAR(presences, trials, suitability, 
spatial.entity, data, n.neighbors, neighbors, suitability.pred=NULL, 
spatial.entity.pred=NULL, burnin = 5000, mcmc = 10000, thin = 10, 
beta.start, Vrho.start, mubeta = 0, Vbeta = 1e+06, priorVrho = 
“1/Gamma”, shape = 0.5, rate = 0.0005, Vrho.max=1000, seed = 1234, 
verbose = 1, save.rho = 0, save.p = 0)
```

Arguments

- `presences`: A vector indicating the number of successes (or presences) for each observation.
- `trials`: A vector indicating the number of trials for each observation. $t_i$ should be superior to zero and superior or equal to $y_i$, the number of successes for observation $i$.
- `suitability`: A one-sided formula of the form $\sim x_1 + ... + x_p$ with $p$ terms specifying the explicative variables for the suitability process.
- `spatial.entity`: A vector indicating the spatial entity identifier (from one to the total number of entities) for each observation. Several observations can occur in one spatial entity. A spatial entity can be a raster cell for example.
- `data`: A data frame containing the model’s variables.
- `n.neighbors`: A vector of integers that indicates the number of neighbors (adjacent entities) of each spatial entity. `length(n.neighbors)` indicates the total number of spatial entities.
- `neighbors`: A vector of integers indicating the neighbors (adjacent entities) of each spatial entity. Must be of the form `c(neighbors of entity 1, neighbors of entity 2, ... , neighbors of the last entity)`. Length of the `neighbors` vector should be equal to `sum(n.neighbors)`.
- `suitability.pred`: An optional data frame in which to look for variables with which to predict. If NULL, the observations are used.
- `spatial.entity.pred`: An optional vector indicating the spatial entity identifier (from one to the total number of entities) for predictions. If NULL, the vector `spatial.entity` for observations is used.
- `burnin`: The number of burnin iterations for the sampler.
- `mcmc`: The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to `burnin+mcmc`. `burnin+mcmc` must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.
- `thin`: The thinning interval used in the simulation. The number of `mcmc` iterations must be divisible by this value.
- `beta.start`: Starting values for $\beta$ parameters of the suitability process. This can either be a scalar or a $p$-length vector.
**Vrho.start** Positive scalar indicating the starting value for the variance of the spatial random effects.

**mubeta** Means of the priors for the $\beta$ parameters of the suitability process. mubeta must be either a scalar or a p-length vector. If mubeta takes a scalar value, then that value will serve as the prior mean for all of the betas. The default value is set to 0 for an uninformative prior.

**Vbeta** Variances of the Normal priors for the $\beta$ parameters of the suitability process. Vbeta must be either a scalar or a p-length vector. If Vbeta takes a scalar value, then that value will serve as the prior variance for all of the betas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

**priorVrho** Type of prior for the variance of the spatial random effects. Can be set to a fixed positive scalar, or to an inverse-gamma distribution ("1/Gamma") with parameters shape and rate, or to a uniform distribution ("Uniform") on the interval $[0, Vrho.max]$. Default set to "1/Gamma".

**shape** The shape parameter for the Gamma prior on the precision of the spatial random effects. Default value is shape=0.05 for uninformative prior.

**rate** The rate (1/scale) parameter for the Gamma prior on the precision of the spatial random effects. Default value is rate=0.0005 for uninformative prior.

**Vrho.max** Upper bound for the uniform prior of the spatial random effect variance. Default set to 1000.

**seed** The seed for the random number generator. Default set to 1234.

**verbose** A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

**save.rho** A switch (0,1) which determines whether or not the sampled values for rhos are saved. Default is 0: the posterior mean is computed and returned in the rho.pred vector. Be careful, setting save.rho to 1 might require a large amount of memory.

**save.p** A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the theta.pred vector. Be careful, setting save.p to 1 might require a large amount of memory.

**Details**

We model an ecological process where the presence or absence of the species is explained by habitat suitability. The ecological process includes an intrinsic conditional autoregressive (iCAR) model for spatial autocorrelation between observations, assuming that the probability of presence of the species at one site depends on the probability of presence of the species on neighboring sites.

**Ecological process:**

$$y_i \sim Binomial(\theta_i, t_i)$$

$$logit(\theta_i) = X_i \beta + \rho_{j(i)}$$

$\rho_j$: spatial random effect

$j(i)$: index of the spatial entity for observation $i$. 
Spatial autocorrelation:
An intrinsic conditional autoregressive model (iCAR) is assumed:

$$\rho_j \sim \text{Normal}(\mu_j, V_\rho/n_j)$$

$\mu_j$: mean of $\rho_{j'}$ in the neighborhood of $j$.

$V_\rho$: variance of the spatial random effects.

$n_j$: number of neighbors for spatial entity $j$.

Value
mcmc
An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance $D$, with $D = -2 \log(\prod_i P(y_i|...))$, is also provided.

rho.pred
If save.rho is set to 0 (default), rho.pred is the predictive posterior mean of the spatial random effect associated to each spatial entity. If save.rho is set to 1, rho.pred is an mcmc object with sampled values for each spatial random effect associated to each spatial entity.

theta.pred
If save.p is set to 0 (default), theta.pred is the predictive posterior mean of the probability associated to the suitability process for each prediction. If save.p is set to 1, theta.pred is an mcmc object with sampled values of the probability associated to the suitability process for each prediction.

theta.latent
Predictive posterior mean of the probability associated to the suitability process for each observation.

Author(s)
Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>

References


See Also
plot.mcmc, summary.mcmc
# Examples

```r
## Not run:

#----------------------------------------------------------------------
# hSDM.binomial.iCAR()
# Example with simulated data
#----------------------------------------------------------------------

#-------------------------
##== Load libraries
library(hSDM)
library(raster)
library(sp)

#-------------------------
##== Multivariate normal distribution
rmvn <- function(n, mu = 0, V = matrix(1), seed=1234) {
  p <- length(mu)
  if (any(is.na(match(dim(V), p)))) {
    stop("Dimension problem!")
  }
  D <- chol(V)
  set.seed(seed)
  t(matrix(rnorm(n*p),ncol=p)%*%D+rep(mu,rep(n,p)))
}

#-------------------------
##== Data simulation

## Set seed for repeatability
seed <- 1234

## Landscape
xLand <- 30
yLand <- 30
Landscape <- raster(ncol=xLand,nrow=yLand,crs='+proj=utm +zone=1')
extent(Landscape) <- c(0,xLand,0,yLand)
coords <- coordinates(Landscape)
cells <- ncell(Landscape)

## Neighbors
neighbors.mat <- adjacent(Landscape, cells=1:ncells, directions=8, pairs=TRUE, sorted=TRUE)
n.neighbors <- as.data.frame(table(as.factor(neighbors.mat[,1])))[,2]
adj <- neighbors.mat[,2]

## Generate symmetric adjacency matrix, A
A <- matrix(0,ncells,ncells)
index.start <- 1
for (i in 1:ncells) {
  index.end <- index.start+n.neighbors[i]-1
  A[index.end+1:index.start,i] <- A[index.end+1:index.start,i] <- 1
  A[index.start:index.end,index.end+1:index.start] <- A[index.start:index.end,index.end+1:index.start] <- 1
  A[index.end+1:index.start,index.end+1:index.start] <- A[index.end+1:index.start,index.end+1:index.start] <- 1
  index.start <- index.end+2
}
```

library(hSDM)

A[i,adj[c(index.start:index.end)]] <- 1
index.start <- index.end+1

# Spatial effects
Vrho.target <- 5
d <- 1 # Spatial dependence parameter = 1 for intrinsic CAR
Q <- diag(n.neighbors)-d*A + diag(.0001,ncells) # Add small constant to make Q non-singular
covrho <- Vrho.target*solve(Q) # Covariance of rhos
set.seed(seed)
rho <- c(rmvn(1,mu=rep(0,ncells),V= covrho,seed=seed)) # Spatial Random Effects
rho <- rho-mean(rho) # Centering rhos on zero

# Raster and plot spatial effects
r.rho <- rasterFromXYZ(cbind(coords,rho))
plot(r.rho)

# Sample the observation sites in the landscape
nsite <- 250
set.seed(seed)
x.coord <- runif(nsite,0,xLand)
set.seed(2*seed)
y.coord <- runif(nsite,0,yLand)
sites.sp <- SpatialPoints(coords=cbind(x.coord,y.coord))
cells <- extract(Landscape,sites.sp,cell=TRUE)[,1]

# Number of visits associated to each observation point
set.seed(seed)
visits <- rpois(nsite,3)
visits[visits==0] <- 1

# Ecological process (suitability)
set.seed(seed)
x1 <- rnorm(nsite,0,1)
set.seed(2*seed)
x2 <- rnorm(nsite,0,1)
X <- cbind(rep(1,nsite),x1,x2)
beta.target <- c(-1,1,-1)
logit.theta <- X %*% beta.target + rho[cells]
theta <- inv.logit(logit.theta)
set.seed(seed)
Y <- rbinom(nsite,visits,theta)

# Relative importance of spatial random effects
RImp <- mean(abs(rho[cells])/abs(X %*% beta.target))
RImp

# Data-sets
data.obs <- data.frame(Y,visits,x1,x2,cell=cells)

# Site-occupancy model
hSDM.binomial.iCAR

Start <- Sys.time() # Start the clock
mod.hSDM.binomial.iCAR <- hSDM.binomial.iCAR(presences=data.obs$Y,
   trials=data.obs$visits,
   suitability=x1+x2,
   spatial.entity=data.obs$cell,
   data=data.obs,
   n.neighbors=n.neighbors,
   neighbors=adj,
   suitability.pred=NULL,
   spatial.entity.pred=NULL,
   burnin=5000, mcmc=5000, thin=5,
   beta.start=0,
   Vrho.start=1,
   mubeta=0, Vbeta=1.0E6,
   priorVrho="1/Gamma",
   shape=0.5, rate=0.0005,
   seed=1234, verbose=1,
   save.rho=1, save.p=0)

Time.hSDM <- difftime(Sys.time(),Start,units="sec") # Time difference

# Computation time
Time.hSDM

#==========
#== Outputs

# Parameter estimates
summary(mod.hSDM.binomial.iCAR$mcmc)
pdf("Posteriors_hSDM.binomial.iCAR.pdf")
plot(mod.hSDM.binomial.iCAR$mcmc)
dev.off()

# Predictions
summary(mod.hSDM.binomial.iCAR$theta.latent)
summary(mod.hSDM.binomial.iCAR$theta.pred)
pdf(file="Pred-Init.pdf")
plot(theta,mod.hSDM.binomial.iCAR$theta.pred)
abline(a=0,b=1,col="red")
dev.off()

# Summary plots for spatial random effects

# rho.pred
rho.pred <- apply(mod.hSDM.binomial.iCAR$rho.pred,2,mean)
r.rho.pred <- rasterFromXYZ(cbind(coords,rho.pred))

# plot
pdf(file="Summary_hSDM.binomial.iCAR.pdf")
par(mfrow=c(2,2))
# rho target
plot(r.rho, main="rho target")
plot(sites.sp,add=TRUE)
# rho estimated
hsdm.Nmixture

Description

The hSDM.Nmixture function can be used to model species distribution including different processes in a hierarchical Bayesian framework: a Poisson suitability process (referring to environmental suitability explaining abundance) and a Binomial observability process (referring to various ecological and methodological issues explaining species detection). The hSDM.Nmixture function calls a Gibbs sampler written in C code which uses an adaptive Metropolis algorithm to estimate the conditional posterior distribution of hierarchical model’s parameters.

Usage

hsdm.Nmixture(# Observations
counts, observability, site, data.observability,
# Habitat
suitability, data.suitability,
# Predictions
suitability.pred = NULL,
# Chains
burnin = 5000, mcmc = 10000, thin = 10,
# Starting values
beta.start,
gamma.start,
# Priors
mubeta = 0, Vbeta = 1.0E6,
mugamma = 0, Vgamma = 1.0E6,
# Various
seed = 1234, verbose = 1,
save.p = 0, save.N = 0)
hSDM.Nmixture

Arguments

- **counts**: A vector indicating the count (or abundance) for each observation.
- **observability**: A one-sided formula of the form $\sim w_1 + \ldots + w_q$ with $q$ terms specifying the explicative variables for the observability process.
- **site**: A vector indicating the site identifier (from one to the total number of sites) for each observation. Several observations can occur at one site. A site can be a raster cell for example.
- **data.observability**: A data frame containing the model’s variables for the observability process.
- **suitability**: A one-sided formula of the form $\sim x_1 + \ldots + x_p$ with $p$ terms specifying the explicative variables for the suitability process.
- **data.suitability**: A data frame containing the model’s variables for the suitability process.
- **suitability.pred**: An optional data frame in which to look for variables with which to predict. If NULL, the observations are used.
- **burnin**: The number of burnin iterations for the sampler.
- **mcmc**: The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to burnin+mcmc. burnin+mcmc must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.
- **thin**: The thinning interval used in the simulation. The number of mcmc iterations must be divisible by this value.
- **beta.start**: Starting values for $\beta$ parameters of the suitability process. This can either be a scalar or a $p$-length vector.
- **gamma.start**: Starting values for $\beta$ parameters of the observability process. This can either be a scalar or a $q$-length vector.
- **mubeta**: Means of the priors for the $\beta$ parameters of the suitability process. mubeta must be either a scalar or a $p$-length vector. If mubeta takes a scalar value, then that value will serve as the prior mean for all of the betas. The default value is set to 0 for an uninformative prior.
- **Vbeta**: Variances of the Normal priors for the $\beta$ parameters of the suitability process. Vbeta must be either a scalar or a $p$-length vector. If Vbeta takes a scalar value, then that value will serve as the prior variance for all of the betas. The default variance is large and set to 1.0E6 for an uninformative flat prior.
- **mugamma**: Means of the Normal priors for the $\gamma$ parameters of the observability process. mugamma must be either a scalar or a $p$-length vector. If mugamma takes a scalar value, then that value will serve as the prior mean for all of the gammas. The default value is set to 0 for an uninformative prior.
- **Vgamma**: Variances of the Normal priors for the $\gamma$ parameters of the observability process. Vgamma must be either a scalar or a $p$-length vector. If Vgamma takes a scalar value, then that value will serve as the prior variance for all of the gammas. The default variance is large and set to 1.0E6 for an uninformative flat prior.
- **seed**: The seed for the random number generator. Default set to 1234.
verbose
A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

save.p
A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the \texttt{lambda.pred} vector. Be careful, setting \texttt{save.p} to 1 might require a large amount of memory.

save.N
A switch (0,1) which determines whether or not the sampled values for the latent count variable N for each observed cells are saved. Default is 0: the mean (rounded to the closest integer) is computed and returned in the \texttt{N.pred} vector. Be careful, setting \texttt{save.N} to 1 might require a large amount of memory.

Details
The model integrates two processes, an ecological process associated to the abundance of the species due to habitat suitability and an observation process that takes into account the fact that the probability of detection of the species is inferior to one.

Ecological process:
\[ N_i \sim \text{Poisson}(\lambda_i) \]
\[ \log(\lambda_i) = X_i \beta \]

Observation process:
\[ y_{it} \sim \text{Binomial}(N_i, \delta_{it}) \]
\[ \text{logit}(\delta_{it}) = W_{it} \gamma \]

Value
\texttt{mcmc}
An \texttt{mcmc} object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance D, with \[ D = -2 \log(\prod_{it} P(y_{it}, N_i | ..)) \], is also provided.

\texttt{lambda.pred}
If \texttt{save.p} is set to 0 (default), \texttt{lambda.pred} is the predictive posterior mean of the abundance associated to the suitability process for each prediction. If \texttt{save.p} is set to 1, \texttt{lambda.pred} is an \texttt{mcmc} object with sampled values of the abundance associated to the suitability process for each prediction.

\texttt{N.pred}
If \texttt{save.N} is set to 0 (default), \texttt{N.pred} is the posterior mean (rounded to the closest integer) of the latent count variable N for each observed cell. If \texttt{save.N} is set to 1, \texttt{N.pred} is an \texttt{mcmc} object with sampled values of the latent count variable N for each observed cell.

\texttt{lambda.latent}
Predictive posterior mean of the abundance associated to the suitability process for each observation.

\texttt{delta.latent}
Predictive posterior mean of the probability associated to the observability process for each observation.

Author(s)
Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>
hSDM.Nmixture

References


See Also

plot.mcmc, summary.mcmc

Examples

```r
## Not run:

#-----------------------------------------
# hSDM.Nmixture()
# Example with simulated data
#-----------------------------------------

#-----------------------------
#== Load libraries
library(hSDM)

#-----------------------------
#== Data simulation

# Number of observation sites
nsite <- 200

# Set seed for repeatability
seed <- 4321

# Ecological process (suitability)
set.seed(seed)
x1 <- rnorm(nsite,0,1)
set.seed(2*seed)
x2 <- rnorm(nsite,0,1)
X <- cbind(rep(1,nsite),x1,x2)
betatarget <- c(-1,1,-1) # Target parameters
log.lambda <- X %*% betatarget
lambda <- exp(log.lambda)
set.seed(seed)
N <- rpois(nsite,lambda)

# Number of visits associated to each observation point
set.seed(seed)
visits <- rpois(nsite,3)
```
visits[visits==0] <- 1
# Vector of observation points
sites <- vector()
for (i in 1:n.site) {
    sites <- c(sites, rep(i, visits[i])
}

# Observation process (detectability)
n.obs <- sum(visits)
set.seed(seed)
w1 <- rnorm(n.obs, 0, 1)
set.seed(2*seed)
w2 <- rnorm(n.obs, 0, 1)
W <- cbind(rep(1, n.obs), w1, w2)
gamma.target <- c(-1, 1, -1) # Target parameters
logit.delta <- W %*% gamma.target
delta <- inv.logit(logit.delta)
set.seed(seed)
Y <- rbinom(n.obs, N[sites], delta)

# Data-sets
data.obs <- data.frame(Y, w1, w2, site=sites)
data.suit <- data.frame(x1, x2)

# Parameter inference with hSDM
Start <- Sys.time() # Start the clock
mod.hSDM.Nmixture <- hsdm.Nmixture(
    counts=data.obs$Y,
    observability=-w1+w2,
    site=data.obs$site,
    data.observability=data.obs,
    site=data.obs$site,
    data.observability=data.obs,
    habitat
    suitability=-x1+x2,
    data.suitability=data.suit,
    predictions
    suitability.ped=NULL,
    chains
    burnin=5000, mcmc=5000, thin=5,
    starting values
    beta.start=0,
    gamma.start=0,
    priors
    mubeta=0, Vbeta=1.0E6,
    mugamma=0, Vgamma=1.0E6,
    various
    seed=1234, verbose=1,
    save=p=0, save.N=1)

Time.hSDM <- difftime(Sys.time(), Start, units="sec") # Time difference

# Computation time
Time.hSDM
### hSDM.Nmixture.iCAR

#### N-mixture model with CAR process

---

**Description**

The hSDM.Nmixture.iCAR function can be used to model species distribution including different processes in a hierarchical Bayesian framework: a Poisson suitability process (referring to environmental suitability explaining abundance) which takes into account the spatial dependence of the
observations, and a *Binomial* observability process (refering to various ecological and methodo-
logical issues explaining the species detection). The hSDM.Nmixture.iCAR function calls a Gibbs
sampler written in C code which uses an adaptive Metropolis algorithm to estimate the conditional
posterior distribution of hierarchical model’s parameters.

Usage

```r
hsDM.Nmixture.iCAR(
  counts, observability, site, data.observability,
  # Habitat
  suitability, data.suitability,
  # Spatial structure
  spatial.entity,
  n.neighbors, neighbors,
  # Predictions
  suitability.pred = NULL, spatial.entity.pred = NULL,
  # Chains
  burnin = 5000, mcmc = 10000, thin = 10,
  # Starting values
  beta.start, gamma.start,
  Vrho.start,
  # Priors
  mubeta = 0, Vbeta = 1.0E6,
  mugamma = 0, Vgamma = 1.0E6,
  priorVrho = "1/Gamma",
  shape = 0.5, rate = 0.0005,
  Vrho.max = 1000,
  # Various
  seed = 123, verbose = 1,
  save.rho = 1, save.p = 0, save.N = 0)
```

Arguments

counts A vector indicating the count (or abundance) for each observation.

observability A one-sided formula of the form $w_1 + \ldots + w_q$ with $q$ terms specifying the
explicative variables for the observability process.

site A vector indicating the site identifier (from one to the total number of sites) for
each observation. Several observations can occur at one site. A site can be a
raster cell for example.

data.observability A data frame containing the model’s variables for the observability process.

suitability A one-sided formula of the form $x_1 + \ldots + x_p$ with $p$ terms specifying the
explicative variables for the suitability process.

data.suitability A data frame containing the model’s variables for the suitability process. The
number of rows of the data frame should be equal to the total number of spatial
entities.
spatial.entity  A vector (of length 'nsite') indicating the spatial entity identifier for each site. Values must be between 1 and the total number of spatial entities. Several sites can be found in one spatial entity. A spatial entity can be a raster cell for example.

n.neighbors  A vector of integers that indicates the number of neighbors (adjacent entities) of each spatial entity. \text{length(n.neighbors)} indicates the total number of spatial entities.

neighbors  A vector of integers indicating the neighbors (adjacent entities) of each spatial entity. Must be of the form c(neighbors of entity 1, neighbors of entity 2, ..., neighbors of the last entity). Length of the neighbors vector should be equal to \text{sum(n.neighbors)}.

suitability.pred  An optional data frame in which to look for variables with which to predict. If NULL, the data frame data.suitability for observations is used.

spatial.entity.pred  An optional vector indicating the spatial entity identifier (from one to the total number of entities) for predictions. If NULL, the vector spatial.entity for observations is used.

burnin  The number of burnin iterations for the sampler.

mcmc  The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to burnin+mcmc. burnin+mcmc must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.

thin  The thinning interval used in the simulation. The number of mcmc iterations must be divisible by this value.

beta.start  Starting values for \( \beta \) parameters of the suitability process. This can either be a scalar or a \( p \)-length vector.

gamma.start  Starting values for \( \beta \) parameters of the observability process. This can either be a scalar or a \( q \)-length vector.

\( \nu_{\rho} \)  Positive scalar indicating the starting value for the variance of the spatial random effects.

mubeta  Means of the priors for the \( \beta \) parameters of the suitability process. mubeta must be either a scalar or a \( p \)-length vector. If mubeta takes a scalar value, then that value will serve as the prior mean for all of the betas. The default value is set to 0 for an uninformative prior.

Vbeta  Variances of the Normal priors for the \( \beta \) parameters of the suitability process. Vbeta must be either a scalar or a \( p \)-length vector. If Vbeta takes a scalar value, then that value will serve as the prior variance for all of the betas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

mugamma  Means of the Normal priors for the \( \gamma \) parameters of the observability process. mugamma must be either a scalar or a \( q \)-length vector. If mugamma takes a scalar value, then that value will serve as the prior mean for all of the gammas. The default value is set to 0 for an uninformative prior.

Vgamma  Variances of the Normal priors for the \( \gamma \) parameters of the observability process. Vgamma must be either a scalar or a \( p \)-length vector. If Vgamma takes a scalar value, then that value will serve as the prior variance for all of the gammas. The default variance is large and set to 1.0E6 for an uninformative flat prior.
priorVrho
Type of prior for the variance of the spatial random effects. Can be set to a fixed positive scalar, or to an inverse-gamma distribution ("1/Gamma") with parameters shape and rate, or to a uniform distribution ("Uniform") on the interval [0, Vrho.max]. Default set to "1/Gamma".

shape
The shape parameter for the Gamma prior on the precision of the spatial random effects. Default value is shape=0.05 for uninformative prior.

rate
The rate (1/scale) parameter for the Gamma prior on the precision of the spatial random effects. Default value is rate=0.0005 for uninformative prior.

Vrho.max
Upper bound for the uniform prior of the spatial random effect variance. Default set to 1000.

seed
The seed for the random number generator. Default set to 1234.

verbose
A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

save.rho
A switch (0,1) which determines whether or not the sampled values for rhos are saved. Default is 0: the posterior mean is computed and returned in the rho.pred vector. Be careful, setting save.rho to 1 might require a large amount of memory.

save.p
A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the lambda.pred vector. Be careful, setting save.p to 1 might require a large amount of memory.

save.N
A switch (0,1) which determines whether or not the sampled values for the latent count variable N for each observed cells are saved. Default is 0: the mean (rounded to the closest integer) is computed and returned in the N.pred vector. Be careful, setting save.N to 1 might require a large amount of memory.

Details
The model integrates two processes, an ecological process associated to the abundance of the species due to habitat suitability and an observation process that takes into account the fact that the probability of detection of the species is inferior to one. The ecological process includes an intrinsic conditional autoregressive model (iCAR) model for spatial autocorrelation between observations, assuming that the abundance of the species at one site depends on the abundance of the species on neighboring sites.

Ecological process:
\[ N_i \sim Poisson(\lambda_i) \]
\[ \log(\lambda_i) = X_i \beta + \rho_i \]
\[ \rho_i: \text{ spatial random effect} \]

Spatial autocorrelation:
An intrinsic conditional autoregressive model (iCAR) is assumed:
\[ \rho_i \sim Normal(\mu_i, V_\rho/n_i) \]
\[ \mu_i: \text{ mean of } \rho_{ij} \text{ in the neighborhood of } i. \]
$V_\rho$: variance of the spatial random effects.

$n_i$: number of neighbors for spatial entity $i$.

**Observation process:**

$$y_{it} \sim \text{Binomial}(N_i, \delta_{it})$$

$$\logit(\delta_{it}) = W_{it} \gamma$$

**Value**

- `mcmc`: An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance $D$, with $D = -2 \log(\prod_{it} P(y_{it}, N_i \mid \ldots))$, is also provided.

- `rho.pred`: If `save.rho` is set to 0 (default), `rho.pred` is the predictive posterior mean of the spatial random effect associated to each spatial entity. If `save.rho` is set to 1, `rho.pred` is an mcmc object with sampled values for each spatial random effect associated to each spatial entity.

- `lambda.pred`: If `save.p` is set to 0 (default), `lambda.pred` is the predictive posterior mean of the abundance associated to the suitability process for each prediction. If `save.p` is set to 1, `lambda.pred` is an mcmc object with sampled values of the abundance associated to the suitability process for each prediction.

- `N.pred`: If `save.N` is set to 0 (default), `N.pred` is the posterior mean (rounded to the closest integer) of the latent count variable $N$ for each observed cell. If `save.N` is set to 1, `N.pred` is an mcmc object with sampled values of the latent count variable $N$ for each observed cell.

- `lambda.latent`: Predictive posterior mean of the abundance associated to the suitability process for each observation.

- `delta.latent`: Predictive posterior mean of the probability associated to the observability process for each observation.

**Author(s)**

Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>

**References**


**See Also**

`plot.mcmc`, `summary.mcmc`
Examples

```r
## Not run:

# hSDM.Nmixture.icAR()
# Example with simulated data

### Load libraries
library(hSDM)
library(raster)
library(sp)

### Multivariate normal distribution
rmvn <- function(n, mu = 0, V = matrix(1), seed=1234) {
  p <- length(mu)
  if (any(is.na(match(dim(V), p)))) {
    stop("Dimension problem!")
  }
  D <- chol(V)
  set.seed(seed)
  t(matrix(rnorm(n*p),ncol=p)%*%D+rep(mu,rep(n,p)))
}

### Data simulation

## Set seed for repeatability
seed <- 4321

## Landscape
xLand <- 20
yLand <- 20
Landscape <- raster(ncol=xLand,nrow=yLand,crs='+proj=utm +zone=1')
extent(Landscape) <- c(0,xLand,0,yLand)
coords <- coordinates(Landscape)
cells <- ncell(Landscape)

## Neighbors
neighbors.mat <- adjacent(Landscape, cells=c(1:ncells), directions=8, pairs=TRUE, sorted=TRUE)
n.neighbors <- as.data.frame(table(as.factor(neighbors.mat[,1])))[,2]
adj <- neighbors.mat[,2]

## Generate symmetric adjacency matrix, A
A <- matrix(0,nCells,nCells)
index.start <- 1
for (i in 1:nCells) {
  for (j in 1:nCells) {
    if (adj[i,j] == 1) {
      A[i,j] <- 1
      A[j,i] <- 1
    }
  }
}
```

index.end <- index.start+n.neighbors[i]-1
A[i,adj[c(index.start:index.end)]] <- 1
index.start <- index.end+1
}

# Spatial effects
Vrho.target <- 5
d <- 1  # Spatial dependence parameter = 1 for intrinsic CAR
Q <- diag(n.neighbors)-d*A + diag(0.0001,n.cells)  # Add small constant to make Q non-singular
covrho <- Vrho.target*solve(Q)  # Covariance of rhos
set.seed(seed)
rho <- c(rmvn(1,rep(0,n.cells),V= covrho,seed=seed))  # Spatial Random Effects
rho <- rho-mean(rho)  # Centering rhos on zero

# Raster and plot spatial effects
r.rho <- rasterFromXYZ(cbind(coords,rho))
plot(r.rho)

# Sample the observation sites in the landscape
nsite <- 150
set.seed(seed)
x coord <- runif(nsite,0,xLand)
set.seed(2*seed)
y coord <- runif(nsite,0,yLand)
sites.sp <- SpatialPoints(coords=cbind(x coord,y coord))
cells <- extract(Landscape,sites.sp,cell=TRUE)[,1]

# Ecological process (suitability)
set.seed(seed)
x1 <- rnorm(nsite,0,1)
set.seed(2*seed)
x2 <- rnorm(nsite,0,1)
X <- cbind(rep(1,nsite),x1,x2)
beta.target <- c(-1,1,-1)
log.lambda <- X *%*% beta.target + rho[cells]
lambda <- exp(log.lambda)
set.seed(seed)
N <- rpois(nsite,lambda)

# Relative importance of spatial random effects
RImp <- mean(abs(rho[cells])/abs(X *%*% beta.target))

# Number of visits associated to each observation point
set.seed(seed)
visits <- rpois(nsite,3)
visits[visits==0] <- 1
# Vector of observation points
sites <- vector()
for (i in 1:nsite) {
  sites <- c(sites,rep(i,visits[i]))
}
# Observation process (detectability)
nobs <- sum(visits)
set.seed(seed)
w1 <- rnorm(nobs,0,1)
set.seed(2*seed)
w2 <- rnorm(nobs,0,1)
W <- cbind(rep(1,nobs),w1,w2)
gamma.target <- c(-1,1,-1)
logit.delta <- W %*% gamma.target
delta <- inv.logit(logit.delta)
set.seed(seed)
Y <- rbinom(nobs,N[sites],delta)

# Data-sets
data.obs <- data.frame(Y,w1,w2,site=sites)
data.suit <- data.frame(x1,x2,cell=cells)

# Parameter inference with hSDM

Start <- Sys.time() # Start the clock
mod.hSDM.Nmixture.ICAR <- hsdm.Nmixture.ICAR(# Observations
counts=data.obs$Y,
observability=-w1+w2,
site=data.obs$site,
data.obs;
# Habitat
suitability=-x1+x2, data.suitability=data.suit,
# Spatial structure
spatial.entity=data.suit$cell,
n.neighbors=n.neighbors, neighbors=adj,
# Predictions
suitability.pred=NULL,
spatial.entity.pred=NULL,
# Chains
burnin=5000, mcmc=5000, thin=5,
# Starting values
beta.start=0,
gamma.start=0,
Vrho.start=1,
# Priors
mubeta=0, Vbeta=1.0E6,
mugamma=0, Vgamma=1.0E6,
priorVrho="1/Gamma",
shape=0.5, rate=0.005,
Vrho.max=10,
# Various
seed=1234, verbose=1,
save.rho=1, save.p=0, save.N=1)
Time.hSDM <- difftime(Sys.time(),Start,units="sec") # Time difference

# Computation time
Time.hSDM
# Parameter estimates
summary(mod.hSDM.Nmixture.iCAR$mcmc)
pdf(file="Posteriors_hSDM.Nmixture.iCAR.pdf")
plot(mod.hSDM.Nmixture.iCAR$mcmc)
dev.off()

# Predictions
summary(mod.hSDM.Nmixture.iCAR$lambda.latent)
summary(mod.hSDM.Nmixture.iCAR$delta.latent)
summary(mod.hSDM.Nmixture.iCAR$lambda.pred)
pdf(file="Pred-Init.pdf")
plot(lambda,mod.hSDM.Nmixture.iCAR$lambda.pred)
abline(a=0,b=1,col="red")
dev.off()

# MCMC for latent variable N
pdf(file="MCMC_N.pdf")
plot(mod.hSDM.Nmixture.iCAR$N.pred)
dev.off()

# Check that Ns are corretly estimated
M <- as.matrix(mod.hSDM.Nmixture.iCAR$N.pred)
N.est <- apply(M,2,mean)
Y.by.site <- tapply(data.obs$Y,data.obs$site,mean) # Mean by site
pdf(file="Check_N.pdf",width=10,height=5)
par(mfrow=c(1,2))
plot(Y.by.site, N.est) ## More individuals are expected (N > Y) due to detection process
abline(a=0,b=1,col="red")
plot(N, N.est) ## N are well estimated
abline(a=0,b=1,col="red")
cor(N, N.est) ## Very close to 1
dev.off()

# Summary plots for spatial random effects

# rho.pred
rho.pred <- apply(mod.hSDM.Nmixture.iCAR$rho.pred,2,mean)
r.rho.pred <- rasterFromXYZ(cbind(coords,rho.pred))

# plot
pdf(file="Summary_hSDM.Nmixture.iCAR.pdf")
par(mfrow=c(2,2))
# rho target
plot(r.rho, main="rho target")
plot(sites.sp,add=TRUE)
# rho estimated
plot(r.rho.pred, main="rho estimated")
# correlation and "shrinkage"
Levels.cells <- sort(unique(cells))
hsdm.Nmixture.K

N-mixture model with K, the maximal theoretical abundance

Description

The hSDM.Nmixture.K function can be used to model species distribution including different processes in a hierarchical Bayesian framework: a Poisson suitability process (referring to environmental suitability explaining abundance) and a Binomial observability process (referring to various ecological and methodological issues explaining species detection). The hSDM.Nmixture.K function calls a Gibbs sampler written in C code which uses an adaptive Metropolis algorithm to estimate the conditional posterior distribution of hierarchical model’s parameters. K is the maximal theoretical abundance sensus Royle 2004.

Usage

hsdm.Nmixture.K(# Observations
counts, observability, site, data.observability,
# Habitat
suitability, data.suitability,
# Predictions
suitability.pred = NULL,
# Chains
burnin = 5000, mcmc = 10000, thin = 10,
# Starting values
beta.start, gamma.start,
# Priors
mubeta = 0, Vbeta = 1.0E6,
mugamma = 0, Vgamma = 1.0E6,
# Various
K,
seed = 1234, verbose = 1,
save.p = 0)

## End(Not run)
**Arguments**

- **counts**
  A vector indicating the count (or abundance) for each observation.

- **observability**
  A one-sided formula of the form \( \sim w_1 + ... + w_q \) with \( q \) terms specifying the explicative variables for the observability process.

- **site**
  A vector indicating the site identifier (from one to the total number of sites) for each observation. Several observations can occur at one site. A site can be a raster cell for example.

- **data.observability**
  A data frame containing the model’s variables for the observability process.

- **suitability**
  A one-sided formula of the form \( \sim x_1 + ... + x_p \) with \( p \) terms specifying the explicative variables for the suitability process.

- **data.suitability**
  A data frame containing the model’s variables for the suitability process.

- **suitability.pred**
  An optional data frame in which to look for variables with which to predict. If NULL, the observations are used.

- **burnin**
  The number of burnin iterations for the sampler.

- **mcmc**
  The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to \( \text{burnin} + \text{mcmc} \). \( \text{burnin} + \text{mcmc} \) must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.

- **thin**
  The thinning interval used in the simulation. The number of mcmc iterations must be divisible by this value.

- **beta.start**
  Starting values for \( \beta \) parameters of the suitability process. This can either be a scalar or a \( p \)-length vector.

- **gamma.start**
  Starting values for \( \beta \) parameters of the observability process. This can either be a scalar or a \( q \)-length vector.

- **mubeta**
  Means of the priors for the \( \beta \) parameters of the suitability process. \( \text{mubeta} \) must be either a scalar or a \( p \)-length vector. If \( \text{mubeta} \) takes a scalar value, then that value will serve as the prior mean for all of the betas. The default value is set to 0 for an uninformative prior.

- **Vbeta**
  Variances of the Normal priors for the \( \beta \) parameters of the suitability process. \( \text{Vbeta} \) must be either a scalar or a \( p \)-length vector. If \( \text{Vbeta} \) takes a scalar value, then that value will serve as the prior variance for all of the betas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

- **mugamma**
  Means of the Normal priors for the \( \gamma \) parameters of the observability process. \( \text{mugamma} \) must be either a scalar or a \( p \)-length vector. If \( \text{mugamma} \) takes a scalar value, then that value will serve as the prior mean for all of the gammas. The default value is set to 0 for an uninformative prior.

- **Vgamma**
  Variances of the Normal priors for the \( \gamma \) parameters of the observability process. \( \text{Vgamma} \) must be either a scalar or a \( p \)-length vector. If \( \text{Vgamma} \) takes a scalar value, then that value will serve as the prior variance for all of the gammas. The default variance is large and set to 1.0E6 for an uninformative flat prior.
Maximal theoretical abundance sensus Royle 2004. It corresponds to the integer upper index of integration for N-mixture. This should be set high enough so that it does not affect the parameter estimates. Note that computation time will increase with K.

seed
The seed for the random number generator. Default set to 1234.

verbose
A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

save.p
A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the lambda.pred vector. Be careful, setting save.p to 1 might require a large amount of memory.

Details
The model integrates two processes, an ecological process associated to the abundance of the species due to habitat suitability and an observation process that takes into account the fact that the probability of detection of the species is inferior to one.

Ecological process:
\[ N_i \sim \text{Poisson}(\lambda_i) \]
\[ \log(\lambda_i) = X_i \beta \]

Observation process:
\[ y_{it} \sim \text{Binomial}(N_i, \delta_{it}) \]
\[ \text{logit}(\delta_{it}) = W_{it} \gamma \]

Value
mcmc
An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance \( D \), with \( D = -2 \log(\prod P(y_{it}, N_i | ...)) \), is also provided.

lambda.pred
If save.p is set to 0 (default), lambda.pred is the predictive posterior mean of the abundance associated to the suitability process for each prediction. If save.p is set to 1, lambda.pred is an mcmc object with sampled values of the abundance associated to the suitability process for each prediction.

lambda.latent
Predictive posterior mean of the abundance associated to the suitability process for each observation.

delta.latent
Predictive posterior mean of the probability associated to the observability process for each observation.

Author(s)
Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>
References


See Also

`plot.mcmc`, `summary.mcmc`

Examples

```r
## Not run:

#== Load libraries
library(hSDM)

#== Data simulation

# Number of observation sites
nsite <- 200

# Set seed for repeatability
seed <- 4321

# Ecological process (suitability)
set.seed(seed)
x1 <- rnorm(nsite,0,1)
set.seed(2*seed)
x2 <- rnorm(nsite,0,1)
X <- cbind(rep(1,nsite),x1,x2)

beta.target <- c(-1,-1,-1) # Target parameters
log.lambda <- X %*% beta.target
lambda <- exp(log.lambda)
set.seed(seed)
N <- rpois(nsite,lambda)

# Number of visits associated to each observation point
set.seed(seed)
visits <- rpois(nsite,3)
```
visits[visits==0] <- 1
# Vector of observation points
sites <- vector()
for (i in 1:nsite) {
  sites <- c(sites, rep(i, visits[i]))
}

# Observation process (detectability)
nobs <- sum(visits)
set.seed(seed)
w1 <- rnorm(nobs, 0, 1)
set.seed(2*seed)
w2 <- rnorm(nobs, 0, 1)
W <- cbind(rep(1, nobs), w1, w2)
gamma.target <- c(-1,1,-1) # Target parameters
logit.delta <- W %*% gamma.target
delta <- inv.logit(logit.delta)
set.seed(seed)
Y <- rbinom(nobs, N[sites], delta)

# Data-sets
data.obs <- data.frame(Y, w1, w2, site=sites)
data.suit <- data.frame(x1, x2)

# Parameter inference with hSDM
Start <- Sys.time() # Start the clock
mod.hSDM.Nmixture.K <- hSDM.Nmixture.K(# Observations
counts=data.obs$Y,
observability=-w1+w2,
site=data.obs$site,
data.observability=data.obs,
# Habitat
suitability=-x1+x2,
data.suitability=data.suit,
# Predictions
suitability.pred=NULL,
# Chains
burnin=5000, mcmc=5000, thin=5,
# Starting values
beta.start=0,
gamma.start=0,
# Priors
mubeta=0, Vbeta=1.0E6,
mugamma=0, Vgamma=1.0E6,
# Various
K=max(data.obs$Y)*2,
seed=1234, verbose=1,
save.p=0)
Time.hSDM <- difftime(Sys.time(), Start, units="sec") # Time difference

# Computation time
hSDM.poisson

---

Poisson log regression model

Description

The hSDM.poisson function performs a Poisson log regression in a Bayesian framework. The function calls a Gibbs sampler written in C code which uses an adaptive Metropolis algorithm to estimate the conditional posterior distribution of model's parameters.

Usage

```r
hsdm.poisson(counts, suitability, data, suitability.pred = NULL,
             burnin = 5000, mcmc = 10000, thin = 10, beta.start, mubeta = 0, Vbeta = 1e+06, seed = 1234, verbose = 1, save.p = 0)
```

Arguments

- **counts**: A vector indicating the count (or abundance) for each observation.
- **suitability**: A one-sided formula of the form `-x1+...+xp` with p terms specifying the explicative covariates for the suitability process of the model.
- **data**: A data frame containing the model's explicative variables.
- **suitability.pred**: An optional data frame in which to look for variables with which to predict. If NULL, the observations are used.
burnin  The number of burnin iterations for the sampler.

mcmc  The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to burnin+mcmc. burnin+mcmc must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.

thin  The thinning interval used in the simulation. The number of mcmc iterations must be divisible by this value.

beta.start  Starting values for beta parameters of the suitability process. If beta.start takes a scalar value, then that value will serve for all of the betas.

mubeta  Means of the priors for the $\beta$ parameters of the suitability process. mubeta must be either a scalar or a p-length vector. If mubeta takes a scalar value, then that value will serve as the prior mean for all of the betas. The default value is set to 0 for an uninformative prior.

vbeta  Variances of the Normal priors for the $\beta$ parameters of the suitability process. vbeta must be either a scalar or a p-length vector. If vbeta takes a scalar value, then that value will serve as the prior variance for all of the betas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

seed  The seed for the random number generator. Default to 1234.

verbose  A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

save.p  A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the lambda.pred vector. Be careful, setting save.p to 1 might require a large amount of memory.

Details

We model the abundance of the species as a function of environmental variables.

Ecological process:

\[ y_i \sim \text{Poisson}(\lambda_i) \]
\[ \log(\lambda_i) = X_i \beta \]

Value

mcmc  An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance $D$, with $D = -2 \log(\prod_i P(y_i | n_i | \beta))$, is also provided.

lambda.pred  If save.p is set to 0 (default), lambda.pred is the predictive posterior mean of the abundance associated to the suitability process for each prediction. If save.p is set to 1, lambda.pred is an mcmc object with sampled values of the abundance associated to the suitability process for each prediction.

lambda.latent  Predictive posterior mean of the abundance associated to the suitability process for each observation.
Author(s)

Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>

References


See Also

`plot.mcmc, summary.mcmc`

Examples

```r
## Not run:

#-----------------------------
# hSDM.poisson()
# Example with simulated data
#-----------------------------

#-----------------------------
## Load libraries
library(hSDM)

#-----------------------------
## Data simulation

## Number of sites
nsite <- 200

## Set seed for repeatability
seed <- 1234

## Ecological process (suitability)
set.seed(seed)
x1 <- rnorm(nsite,0,1)
set.seed(2*seed)
x2 <- rnorm(nsite,0,1)
X <- cbind(rep(1,nsite),x1,x2)
beta.target <- c(-1,1,-1)
log.lambda <- X %*% beta.target
lambda <- exp(log.lambda)
set.seed(seed)
Y <- rpois(nsite,lambda)

## Data-sets
```
data.obs <- data.frame(Y,x1,x2)

#========================================
#== Site-occupancy model

mod.hSDM.poisson <- hSDM.poisson(counts=data.obs$Y,
suitability=x1+x2,
data=data.obs,
suitability.pred=NULL,
burnin=1000, mcmc=1000, thin=1,
beta.start=0,
mubeta=0, Vbeta=1.0E6,
seed=1234, verbose=1,
save.p=0)

#=====
#== Outputs

#= Parameter estimates
summary(mod.hSDM.poisson$mcmc)
pdf(file="Posterior_hSDM.poisson.pdf")
plot(mod.hSDM.poisson$mcmc)
dev.off()

#== glm resolution to compare
mod.glm <- glm(Y~x1+x2,family="poisson",data=data.obs)
summary(mod.glm)

#= Predictions
summary(mod.hSDM.poisson$lambda.latent)
summary(mod.hSDM.poisson$lambda.pred)
pdf(file="Pred-Init.pdf")
plot(lambda,mod.hSDM.poisson$lambda.pred)
abline(a=0,b=1,col="red")
dev.off()

## End(Not run)

---

**hSDM.poisson.iCAR**  
**Poisson log regression model with CAR process**

---

**Description**

The `hSDM.poisson.iCAR` function performs a Poisson log regression in a hierarchical Bayesian framework. The suitability process includes a spatial correlation process. The spatial correlation is modelled using an intrinsic CAR model. The `hSDM.poisson.iCAR` function calls a Gibbs sampler written in C code which uses an adaptive Metropolis algorithm to estimate the conditional posterior distribution of hierarchical model’s parameters.
Usage

hSDM.poisson.iCAR(counts, suitability, spatial.entity, data, 
n.neighbors, neighbors, suitability.pred=NULL, spatial.entity.pred=NULL, 
burnin = 5000, mcmc = 10000, thin = 10, beta.start, Vrho.start, mubeta = 
0, Vbeta = 1e+06, priorVrho = “1/Gamma”, shape = 0.5, rate = 0.0005, 
Vrho.max=1000, seed = 1234, verbose = 1, save.rho = 0, save.p = 0)

Arguments

counts A vector indicating the count (or abundance) for each observation.
suitability A one-sided formula of the form \( \sim x_1 + \ldots + x_p \) with \( p \) terms specifying the explicative variables for the suitability process.
spatial.entity A vector indicating the spatial entity identifier (from one to the total number of entities) for each observation. Several observations can occur in one spatial entity. A spatial entity can be a raster cell for example.
data A data frame containing the model’s variables.
n.neighbors A vector of integers that indicates the number of neighbors (adjacent entities) of each spatial entity. \text{length}(\text{n.neighbors}) \text{ indicates the total number of spatial entities.}
neighbors A vector of integers indicating the neighbors (adjacent entities) of each spatial entity. Must be of the form (neighbors of entity 1, neighbors of entity 2, ..., neighbors of the last entity). Length of the neighbors vector should be equal to \text{sum(n.neighbors)}.
suitability.pred An optional data frame in which to look for variables with which to predict. If NULL, the observations are used.
spatial.entity.pred An optional vector indicating the spatial entity identifier (from one to the total number of entities) for predictions. If NULL, the vector spatial.entity for observations is used.
burnin The number of burnin iterations for the sampler.
mcmc The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to burnin+mcmc. burnin+mcmc must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.
thin The thinning interval used in the simulation. The number of mcmc iterations must be divisible by this value.
beta.start Starting values for \( \beta \) parameters of the suitability process. This can either be a scalar or a \( p \)-length vector.
Vrho.start Positive scalar indicating the starting value for the variance of the spatial random effects.
mubeta Means of the priors for the \( \beta \) parameters of the suitability process. \text{mubeta} must be either a scalar or a \( p \)-length vector. If \text{mubeta} takes a scalar value, then that value will serve as the prior mean for all of the betas. The default value is set to 0 for an uninformative prior.
Vbeta

Variances of the Normal priors for the $\beta$ parameters of the suitability process. Vbeta must be either a scalar or a p-length vector. If Vbeta takes a scalar value, then that value will serve as the prior variance for all of the betas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

priorVrho

Type of prior for the variance of the spatial random effects. Can be set to a fixed positive scalar, or to an inverse-gamma distribution ("1/Gamma") with parameters shape and rate, or to a uniform distribution ("Uniform") on the interval [0, Vrho.max]. Default set to "1/Gamma".

shape

The shape parameter for the Gamma prior on the precision of the spatial random effects. Default value is shape=0.05 for uninformative prior.

rate

The rate (1/scale) parameter for the Gamma prior on the precision of the spatial random effects. Default value is rate=0.0005 for uninformative prior.

Vrho.max

Upper bound for the uniform prior of the spatial random effect variance. Default set to 1000.

seed

The seed for the random number generator. Default set to 1234.

verbose

A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

save.rho

A switch (0,1) which determines whether or not the sampled values for rhos are saved. Default is 0: the posterior mean is computed and returned in the rho.pred vector. Be careful, setting save.rho to 1 might require a large amount of memory.

save.p

A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the lambda.pred vector. Be careful, setting save.p to 1 might require a large amount of memory.

Details

We model an ecological process where the abundance of the species is explained by habitat suitability. The ecological process includes an intrinsic conditional autoregressive (iCAR) model for spatial autocorrelation between observations, assuming that the probability of presence of the species at one site depends on the probability of presence of the species on neighboring sites.

**Ecological process:**

$$y_i \sim \text{Poisson}(\lambda_i, t_i)$$

$$\log(\lambda_i) = X_i \beta + \rho_{j(i)}$$

$\rho_{j}$: spatial random effect

$j(i)$: index of the spatial entity for observation $i$.

**Spatial autocorrelation:**

An intrinsic conditional autoregressive model (iCAR) is assumed:

$$\rho_j \sim \text{Normal}(\mu_j, V_{\rho}/n_j)$$

$\mu_j$: mean of $\rho_{j'}$ in the neighborhood of $j$.

$V_{\rho}$: variance of the spatial random effects.

$n_j$: number of neighbors for spatial entity $j$. 
Value

- **mcmc**
  An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance $D$, with $D = -2 \log(\prod_i P(y_i|...))$, is also provided.

- **rho.pred**
  If `save.rho` is set to 0 (default), `rho.pred` is the predictive posterior mean of the spatial random effect associated to each spatial entity. If `save.rho` is set to 1, `rho.pred` is an mcmc object with sampled values for each spatial random effect associated to each spatial entity.

- **lambda.pred**
  If `save.lambda` is set to 0 (default), `lambda.pred` is the predictive posterior mean of the abundance associated to the suitability process for each prediction. If `save.lambda` is set to 1, `lambda.pred` is an mcmc object with sampled values of the abundance associated to the suitability process for each prediction.

- **lambda.latent**
  Predictive posterior mean of the abundance associated to the suitability process for each observation.

Author(s)

Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>

References


See Also

- plot.mcmc, summary.mcmc

Examples

```r
## Not run:

#---------------------------------------------------------------
# hSDM.poisson.iCAR()
# Example with simulated data
#---------------------------------------------------------------

#----------------------
#== Load libraries
library(hSDM)
```
library(raster)
library(sp)

#-----------------------------
#== Multivariate normal distribution
rmvn <- function(n, mu = 0, V = matrix(1), seed=1234) {
  p <- length(mu)
  if (any(is.na(match(dim(V), p)))) {
    stop("Dimension problem!")
  }
  D <- chol(V)
  set.seed(seed)
  t(matrix(rnorm(n*p),ncol=p)%*%D+rep(mu,rep(n,p)))
}

#-----------------------------
#== Data simulation

# Set seed for repeatability
seed <- 1234

# Landscape
xLand <- 30
yLand <- 30
landscape <- raster(ncol=xLand,nrow=yLand,crs='+proj=utm +zone=1')
landscape[] <- 0
extent(landscape) <- c(0,xLand,0,yLand)
coords <- coordinates(landscape)
cells <- ncell(landscape)

# Neighbors
neighbors.mat <- adjacent(landscape, cells=c(1:ncells), directions=8, pairs=TRUE, sorted=TRUE)
n.neighbors <- as.data.frame(table(as.factor(neighbors.mat[,1])))[,2]
adj <- neighbors.mat[,2]

# Generate symmetric adjacency matrix, A
A <- matrix(0,ncells,ncells)
index.start <- 1
for (i in 1:ncells) {
  index.end <- index.start+n.neighbors[i]-1
  A[i,adj[c(index.start:index.end)]] <- 1
  index.start <- index.end+1
}

# Spatial effects
Vrho.target <- 5
d <- 1  # Spatial dependence parameter = 1 for intrinsic CAR
Q <- diag(n.neighbors)-d*A + diag(.0001,ncells)  # Add small constant to make Q non-singular
covrho <- Vrho.target*solve(Q)  # Covariance of rhos
set.seed(seed)
rho <- c(rmvn(1,mu=rep(0,ncells),V=covrho,seed=seed))  # Spatial Random Effects
rho <- rho-mean(rho)  # Centering rhos on zero
```r
# Raster and plot spatial effects
r.rho <- rasterFromXYZ(cbind(coords,rho))
plot(r.rho)

# Sample the observation sites in the landscape
nsite <- 250
set.seed(seed)
x.coord <- runif(nsite,0,x.land)
set.seed(2*seed)
y.coord <- runif(nsite,0,y.land)
sites.sp <- SpatialPoints(coords=cbind(x.coord,y.coord))
cells <- extract(Landscape,sites.sp,cell=TRUE)[,1]

# Ecological process (suitability)
set.seed(seed)
x1 <- rnorm(nsite,0,1)
set.seed(2*seed)
x2 <- rnorm(nsite,0,1)
X <- cbind(rep(1,nsite),x1,x2)
beta.target <- c(-1,1,-1)
log.lambda <- X * beta.target + rho[cells]
lambda <- exp(log.lambda)
set.seed(seed)
Y <- rpois(nsite,lambda)

# Relative importance of spatial random effects
RImp <- mean(abs(rho[cells]) / abs(X %*% beta.target))
RImp

# Data-sets
data.obs <- data.frame(Y,x1,x2,cell=cells)

# Site-occupancy model

Start <- Sys.time() # Start the clock
mod.hSDM.poisson.iCAR <- hSDM.poisson.iCAR(counts=data.obs$Y,
suitability=x1+x2,
spatial.entity=data.obs$cell,
data=data.obs,
n.neighbors=n.neighbors,
neighbors=adj,
suitability.pred=NULL,
spatial.entity.pred=NULL,
burnin=5000, mcmc=5000, thin=5,
beta.start=0,
Vrho.start=1,
mubeta=0, Vbeta=1.0E6,
priorVrho="1/Gamma",
shape=0.5, rate=0.0005,
seed=1234, verbose=1,
save.rho=1, save.p=0)

Time.hSDM <- difftime(Sys.time(),Start,units="sec") # Time difference
```
# Computation time
Time.hSDM

# Outputs

# Parameter estimates
summary(mod.hSDM.poisson.iCAR$mcmc)
pdf("Posterior_hSDM.poisson.iCAR.pdf")
plot(mod.hSDM.poisson.iCAR$mcmc)
dev.off()

# Predictions
summary(mod.hSDM.poisson.iCAR$lambda.latent)
summary(mod.hSDM.poisson.iCAR$lambda.pred)
pdf(file="Pred-Init.pdf")
plot(lambda, mod.hSDM.poisson.iCAR$lambda.pred)
abline(a=0, b=1, col="red")
dev.off()

# Summary plots for spatial random effects

# rho.pred
rho.pred <- apply(mod.hSDM.poisson.iCAR$rho.pred, 2, mean)
r.rho.pred <- rasterFromXYZ(cbind(coords, rho.pred))

# plot
pdf(file="Summary_hSDM.poisson.iCAR.pdf")
par(mfrow=c(2,2))
# rho target
plot(r.rho, main="rho target")
plot(sites.sp, add=TRUE)
# rho estimated
plot(r.rho.pred, main="rho estimated")
# correlation and "shrinkage"
Levels.cells <- sort(unique(cells))
plot(rho[-Levels.cells], rho.pred[-Levels.cells],
     xlim=range(rho),
     ylim=range(rho),
     xlab="rho target",
     ylab="rho estimated")
points(rho[Levels.cells], rho.pred[Levels.cells], pch=16, col="blue")
legend(x=-3, y=4, legend="Visited cells", col="blue", pch=16, bty="n")
abline(a=0, b=1, col="red")
dev.off()

## End(Not run)
**Description**

The `hSDM.siteocc` function can be used to model species distribution including different processes in a hierarchical Bayesian framework: a *Bernoulli* suitability process (refering to environmental suitability) and a *Bernoulli* observability process (refering to various ecological and methodological issues explaining the species detection). The `hSDM.siteocc` function calls a Gibbs sampler written in C code which uses a Metropolis algorithm to estimate the conditional posterior distribution of hierarchical model’s parameters.

**Usage**

```r
hSDM.siteocc(# Observations
presence, observability, site, data.observability,
# Habitat
suitability, data.suitability, # Predictions
suitability.pred = NULL, # Chains
burnin = 1000, mcmc = 1000, thin = 1,
# Starting values
beta.start,
gamma.start, # Priors
mubeta = 0, Vbeta = 1.0E6,
mugamma = 0, Vgamma = 1.0E6,
# Various
seed = 1234, verbose = 1,
save.p = 0)
```

**Arguments**

- **presence** A vector indicating the presence/absence for each observation.
- **observability** A one-sided formula of the form $\sim w_1 + ... + w_q$ with $q$ terms specifying the explicative variables for the observability process.
- **site** A vector indicating the site identifier (from one to the total number of sites) for each observation. Several observations can occur at one site. A site can be a raster cell for example.
- **data.observability** A data frame containing the model’s variables for the observability process.
- **suitability** A one-sided formula of the form $\sim x_1 + ... + x_p$ with $p$ terms specifying the explicative variables for the suitability process.
- **data.suitability** A data frame containing the model’s variables for the suitability process.
suitability.pred

An optional data frame in which to look for variables with which to predict. If NULL, the observations are used.

burnin

The number of burnin iterations for the sampler.

mcmc

The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to burnin+mcmc. burnin+mcmc must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.

thin

The thinning interval used in the simulation. The number of mcmc iterations must be divisible by this value.

beta.start

Starting values for $\beta$ parameters of the suitability process. This can either be a scalar or a $p$-length vector.

gamma.start

Starting values for $\beta$ parameters of the observability process. This can either be a scalar or a $q$-length vector.

mubeta

Means of the priors for the $\beta$ parameters of the suitability process. mubeta must be either a scalar or a $p$-length vector. If mubeta takes a scalar value, then that value will serve as the prior mean for all of the betas. The default value is set to 0 for an uninformative prior.

Vbeta

Variances of the Normal priors for the $\beta$ parameters of the suitability process. Vbeta must be either a scalar or a $p$-length vector. If Vbeta takes a scalar value, then that value will serve as the prior variance for all of the betas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

mugamma

Means of the Normal priors for the $\gamma$ parameters of the observability process. mugamma must be either a scalar or a $p$-length vector. If mugamma takes a scalar value, then that value will serve as the prior mean for all of the gammas. The default value is set to 0 for an uninformative prior.

Vgamma

Variances of the Normal priors for the $\gamma$ parameters of the observability process. Vgamma must be either a scalar or a $p$-length vector. If Vgamma takes a scalar value, then that value will serve as the prior variance for all of the gammas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

seed

The seed for the random number generator. Default set to 1234.

verbose

A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

save.p

A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the lambda.pred vector. Be careful, setting save.p to 1 might require a large amount of memory.

Details

The model integrates two processes, an ecological process associated to the presence or absence of the species due to habitat suitability and an observation process that takes into account the fact that the probability of detection of the species is inferior to one.

Ecological process:

$$z_i \sim Bernoulli(\theta_i)$$
\[
\text{logit}(\theta_i) = X_i \beta
\]

**Observation process:**

\[
y_{it} \sim \text{Bernoulli}(z_i \delta_{it})
\]

\[
\text{logit}(\delta_{it}) = W_{it} \gamma
\]

**Value**

- `mcmc` An `mcmc` object that contains the posterior sample. This object can be summarized by functions provided by the `coda` package. The posterior sample of the deviance `D`, with \(D = -2 \log(\prod_{it} P(y_{it}, N_{it} | ...))\), is also provided.

- `theta.pred` If `save.p` is set to 0 (default), `theta.pred` is the predictive posterior mean of the probability associated to the suitability process for each prediction. If `save.p` is set to 1, `theta.pred` is an `mcmc` object with sampled values of the probability associated to the suitability process for each prediction.

- `theta.latent` Predictive posterior mean of the probability associated to the suitability process for each site.

- `delta.latent` Predictive posterior mean of the probability associated to the observability process for each observation.

**Author(s)**

Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>

**References**


**See Also**

`plot.mcmc`, `summary.mcmc`

**Examples**

```r
## Not run:

# The suiteocc function
# hSDM.siteocc()
# Example with simulated data
#----------------------------------------------------------

#----------------------
#== Load libraries
library(hSDM)
#----------------------
#== Data simulation
```
# Number of observation sites
nsite <- 200

# Set seed for repeatability
seed <- 4321

# Ecological process (suitability)
set.seed(seed)
x1 <- rnorm(nsite, 0, 1)
set.seed(2 * seed)
x2 <- rnorm(nsite, 0, 1)
X <- cbind(rep(1, nsite), x1, x2)
beta.target <- c(-1, 1, -1) # Target parameters
logit.theta <- X %*% beta.target
theta <- inv.logit(logit.theta)
set.seed(seed)
Z <- rbinom(nsite, 1, theta)

# Number of visits associated to each observation point
set.seed(seed)
visits <- rpois(nsite, 3)
visits[visits == 0] <- 1
# Vector of observation points
sites <- vector()
for (i in 1:nsite) {
  sites <- c(sites, rep(i, visits[i]))
}

# Observation process (detectability)
nobs <- sum(visits)
set.seed(seed)
w1 <- rnorm(nobs, 0, 1)
set.seed(2 * seed)
w2 <- rnorm(nobs, 0, 1)
W <- cbind(rep(1, nobs), w1, w2)
gamma.target <- c(-1, 1, -1) # Target parameters
logit.delta <- W %*% gamma.target
delta <- inv.logit(logit.delta)
set.seed(seed)
Y <- rbinom(nobs, 1, delta * Z[sites])

# Data-sets
data.obs <- data.frame(Y, w1, w2, site = sites)
data.suit <- data.frame(x1, x2)

# Parameter inference with hSDM
Start <- Sys.time() # Start the clock
mod.hSDM.siteocc <- hSDM.siteocc(# Observations
  presence = data.obs$Y,
observability=-w1+w2,
site=data.obs$site,
data.obs$observability=data.obs,
  # Habitat
suitability=x1+x2,
data.suitability=data.suit,
  # Predictions
suitability.pred=NULL,
  # Chains
burnin=2000, mcmc=2000, thin=2,
  # Starting values
beta.start=0,
gamma.start=0,
  # Priors
mubeta=0, Vbeta=1.0E6,
mugamma=0, Vgamma=1.0E6,
  # Various
seed=1234, verbose=1, save.p=0)
Time.hSDM <- difftime(Sys.time(),Start.units="sec") # Time difference

# Computation time
Time.hSDM

#==========
# Outputs

# Parameter estimates
summary(mod.hSDM.siteocc$mcmc)
pdf(file="Posterior_hSDM.siteocc.pdf")
plot(mod.hSDM.siteocc$mcmc)
dev.off()

# Predictions
summary(mod.hSDM.siteocc$theta.latent)
summary(mod.hSDM.siteocc$delta.latent)
summary(mod.hSDM.siteocc$theta.pred)
pdf(file="Pred-Init.pdf")
plot(theta,mod.hSDM.siteocc$theta.pred)
abline(a=0,b=1,col="red")
dev.off()

## End(Not run)
Description

The hSDM.siteocc.iCAR function can be used to model species distribution including different processes in a hierarchical Bayesian framework: a Bernoulli suitability process (referring to environmental suitability) which takes into account the spatial dependence of the observations, and a Bernoulli observability process (referring to various ecological and methodological issues explaining the species detection). The hSDM.siteocc.iCAR function calls a Gibbs sampler written in C code which uses an adaptive Metropolis algorithm to estimate the conditional posterior distribution of hierarchical model’s parameters.

Usage

hSDM.siteocc.iCAR(# Observations

presence, observability, site, data.observability,
  # Habitat
suitability, data.suitability,
  # Spatial structure
spatial.entity, n.neighbors, neighbors,
  # Predictions
suitability.pred = NULL, spatial.entity.pred = NULL,
  # Chains
burnin = 1000, mcmc = 1000, thin = 1,
  # Starting values
beta.start, gamma.start, Vrho.start,
  # Priors
mubeta = 0, Vbeta = 1.0E6,
mugamma = 0, Vgamma = 1.0E6,
priorVrho = "1/Gamma",
shape = 0.5, rate = 0.0005,
Vrho.max = 1000,
  # Various
seed = 1234, verbose = 1,
  save.rho = 0, save.p = 0)

Arguments

presence A vector indicating the presence/absence for each observation.
observability A one-sided formula of the form $\sim w_1 + ... + w_q$ with $q$ terms specifying the explicative variables for the observability process.
site A vector indicating the site identifier (from one to the total number of sites) for each observation. Several observations can occur at one site. A site can be a raster cell for example.
data.observability A data frame containing the model’s variables for the observability process.
suitability A one-sided formula of the form $\sim x_1 + ... + x_p$ with $p$ terms specifying the explicative variables for the suitability process.
data.suitability  
A data frame containing the model’s variables for the suitability process.

spatial.entity  
A vector (of length 'n.site') indicating the spatial entity identifier for each site. Values must be between 1 and the total number of spatial entities. Several sites can be found in one spatial entity. A spatial entity can be a raster cell for example.

n.neighbors  
A vector of integers that indicates the number of neighbors (adjacent entities) of each spatial entity. length(n.neighbors) indicates the total number of spatial entities.

neighbors  
A vector of integers indicating the neighbors (adjacent entities) of each spatial entity. Must be of the form c(neighbors of entity 1, neighbors of entity 2, ... , neighbors of the last entity). Length of the neighbors vector should be equal to sum(n.neighbors).

suitability.pred  
An optional data frame in which to look for variables with which to predict. If NULL, the observations are used.

spatial.entity.pred  
An optional vector indicating the spatial entity identifier (from one to the total number of entities) for predictions. If NULL, the vector spatial.entity for observations is used.

burnin  
The number of burnin iterations for the sampler.

mcmc  
The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to burnin+mcmc. burnin+mcmc must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.

thin  
The thinning interval used in the simulation. The number of mcmc iterations must be divisible by this value.

beta.start  
Starting values for $\beta$ parameters of the suitability process. This can either be a scalar or a $p$-length vector.

gamma.start  
Starting values for $\gamma$ parameters of the observability process. This can either be a scalar or a $q$-length vector.

vrho.start  
Positive scalar indicating the starting value for the variance of the spatial random effects.

mubeta  
Means of the priors for the $\beta$ parameters of the suitability process. mubeta must be either a scalar or a $p$-length vector. If mubeta takes a scalar value, then that value will serve as the prior mean for all of the betas. The default value is set to 0 for an uninformative prior.

Vbeta  
Variances of the Normal priors for the $\beta$ parameters of the suitability process. Vbeta must be either a scalar or a $p$-length vector. If Vbeta takes a scalar value, then that value will serve as the prior variance for all of the betas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

mugamma  
Means of the Normal priors for the $\gamma$ parameters of the observability process. mugamma must be either a scalar or a $p$-length vector. If mugamma takes a scalar value, then that value will serve as the prior mean for all of the gammas. The default value is set to 0 for an uninformative prior.
Vgamma | Variances of the Normal priors for the $\gamma$ parameters of the observability process. Vgamma must be either a scalar or a p-length vector. If Vgamma takes a scalar value, then that value will serve as the prior variance for all of the gammas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

priorVrho | Type of prior for the variance of the spatial random effects. Can be set to a fixed positive scalar, or to an inverse-gamma distribution ("1/Gamma") with parameters shape and rate, or to a uniform distribution ("Uniform") on the interval [0, Vrho.max]. Default set to "1/Gamma".

shape | The shape parameter for the Gamma prior on the precision of the spatial random effects. Default value is shape=0.05 for uninformative prior.

rate | The rate (1/scale) parameter for the Gamma prior on the precision of the spatial random effects. Default value is rate=0.0005 for uninformative prior.

Vrho.max | Upper bound for the uniform prior of the spatial random effect variance. Default set to 1000.

seed | The seed for the random number generator. Default set to 1234.

verbose | A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

save.rho | A switch (0,1) which determines whether or not the sampled values for rhos are saved. Default is 0: the posterior mean is computed and returned in the rho.pred vector. Be careful, setting save.rho to 1 might require a large amount of memory.

save.p | A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the theta.pred vector. Be careful, setting save.p to 1 might require a large amount of memory.

Details
The model integrates two processes, an ecological process associated to the presence or absence of the species due to habitat suitability and an observation process that takes into account the fact that the probability of detection of the species is inferior to one. The ecological process includes an intrinsic conditional autoregressive model (iCAR) model for spatial autocorrelation between observations, assuming that the probability of presence of the species at one site depends on the probability of presence of the species on neighboring sites.

Ecological process:

$$z_i \sim Bernoulli(\theta_i)$$

$$\logit(\theta_i) = X_i \beta + \rho_j(i)$$

$\rho_j$: spatial random effect

$j(i)$: index of the spatial entity for observation $i$.

Spatial autocorrelation:

An intrinsic conditional autoregressive model (iCAR) is assumed:

$$\rho_j \sim Normal(\mu_j, V_{\rho}/n_j)$$
\(\mu_j\): mean of \(\rho_j'\) in the neighborhood of \(j\).

\(V_j\): variance of the spatial random effects.

\(n_j\): number of neighbors for spatial entity \(j\).

**Observation process:**

\[ y_{it} \sim \text{Bernoulli}(z_i \ast \delta_{it}) \]

\[ \text{logit}(\delta_{it}) = W_{it} \gamma \]

**Value**

- `mcmc`: An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance \(D\), with \(D = -2 \log(\prod_i P(y_i, z_i|...))\), is also provided.

- `rho.pred`: If `save.rho` is set to 0 (default), `rho.pred` is the predictive posterior mean of the spatial random effect associated to each spatial entity. If `save.rho` is set to 1, `rho.pred` is an mcmc object with sampled values for each spatial random effect associated to each spatial entity.

- `theta.pred`: If `save.p` is set to 0 (default), `theta.pred` is the predictive posterior mean of the probability associated to the suitability process for each prediction. If `save.p` is set to 1, `theta.pred` is an mcmc object with sampled values of the probability associated to the suitability process for each prediction.

- `theta.latent`: Predictive posterior mean of the probability associated to the suitability process for each site.

- `delta.latent`: Predictive posterior mean of the probability associated to the observability process for each observation.

**Author(s)**

Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>

**References**


See Also

plot.mcmc, summary.mcmc

Examples

```r
## Not run:

#==============================================================
# hSDM.siteocc.iCAR()
# Example with simulated data
#==============================================================

#== Load libraries
library(hSDM)
library(raster)
library(sp)

#==============================================================
#== Multivariate normal distribution
rmvn <- function(n, mu = 0, V = matrix(1), seed=1234) {
  p <- length(mu)
  if (any(is.na(match(dim(V), p)))) {
    stop("Dimension problem!"")
  }
  D <- chol(V)
  set.seed(seed)
  t(matrix(rnorm(n*p),ncol=p)%*%D+rep(mu,rep(n,p)))
}

#== Data simulation

#== Set seed for repeatability
seed <- 1234

#== Landscape
xLand <- 30
yLand <- 30
Landscape <- raster(ncol=xLand,nrow=yLand,crs='+proj=utm +zone=1')
extent(Landscape) <- c(0,xLand,0,yLand)
coords <- coordinates(Landscape)
ncells <- ncell(Landscape)

#== Neighbors
neighbors.mat <- adjacent(Landscape, cells=c(1:ncells), directions=8, pairs=TRUE, sorted=TRUE)
n.neighbors <- as.data.frame(table(as.factor(neighbors.mat[,1])))[,2]
adj <- neighbors.mat[,2]

#== Generate symmetric adjacency matrix, A
```
A <- matrix(0, ncells, ncells)
index.start <- 1
for (i in 1:ncells) {
  index.end <- index.start + n.neighbors[i] - 1
  A[i, adj[c(index.start:index.end)]] <- 1
  index.start <- index.end + 1
}

# Spatial effects
Vrho.target <- 5
d <- 1 # Spatial dependence parameter = 1 for intrinsic CAR
Q <- diag(n.neighbors) - d*A + diag(.0001, ncells) # Add small constant to make Q non-singular
covrho <- Vrho.target*solve(Q) # Covariance of rhos
set.seed(seed)
rho <- c(rmvn(1, mu=rep(0, ncells), V=covrho, seed=seed)) # Spatial Random Effects
rho <- rho-mean(rho) # Centering rhos on zero

# Raster and plot spatial effects
r.rho <- rasterFromXYZ(cbind(coords, rho))
plot(r.rho)

# Sample the observation sites in the landscape
nsite <- 250
set.seed(seed)
x.coord <- runif(nsite, 0, xLand)
set.seed(2*seed)
y.coord <- runif(nsite, 0, yLand)
sites.sp <- SpatialPoints(coords=cbind(x.coord, y.coord))
cells <- extract(Landscape, sites.sp, cell=TRUE)[,1]

# Ecological process (suitability)
set.seed(seed)
x1 <- rnorm(nsite, 0, 1)
set.seed(2*seed)
x2 <- rnorm(nsite, 0, 1)
X <- cbind(rep(1, nsite), x1, x2)
beta.target <- c(-1, 1, -1)
logit.theta <- X %*% beta.target + rho[cells]
theta <- inv.logit(logit.theta)
set.seed(seed)
Z <- rbinom(nsite, 1, theta)

# Relative importance of spatial random effects
RImp <- mean(abs(rho[cells])/abs(X %*% beta.target))
RImp

# Number of visits associated to each observation point
set.seed(seed)
visits <- rpois(nsite, 3)
visits[visits==0] <- 1
# Vector of observation points
sites <- vector()
for (i in 1:nsite) {
sites <- c(sites,rep(i,visits[i]))
#
# Observation process (detectability)
# nobs <- sum(visits)
set.seed(seed)
w1 <- rnorm(nobs,0,1)
set.seed(2*seed)
w2 <- rnorm(nobs,0,1)
W <- cbind(rep(1,nobs),w1,w2)
gamma.target <- c(-1,1,-1)
logit.delta <- W %*% gamma.target
delta <- inv.logit(logit.delta)
set.seed(seed)
Y <- rbinom(nobs,1,delta*Z[sites])
#
# Data-sets
data.obs <- data.frame(Y,w1,w2,site=sites)
data.suit <- data.frame(x1,x2,cell=cells)

# Parameter inference with hSDM
Start <- Sys.time() # Start the clock
mod.hSDM.siteocc.iCAR <- hSDM.siteocc.iCAR(# Observations
  presence=data.obs$Y,
  observability=-w1+w2,
  site=data.obs$site,
  data.observability=data.obs,
  # Habitat
  suitability=-x1+x2, data.suitability=data.suit,
  # Spatial structure
  spatial.entity=data.suit$cell,
  n.neighbors=n.neighbors, neighbors=adj,
  # Predictions
  suitability.ped=NULL,
  spatial.entity.ped=NULL,
  # Chains
  burnin=10000, mcmc=5000, thin=5,
  # Starting values
  beta.start=0,
  gamma.start=0,
  Vrho.start=1,
  # Priors
  mubeta=0, Vbeta=1.0E6,
  mugamma=0, Vgamma=1.0E6,
  priorVrho="Uniform",
  Vrho.max=10,
  # Various
  seed=1234, verbose=1,
  save.rho=1, save.p=0)
Time.hSDM <- difftime(Sys.time(),Start,units="sec") # Time difference
hSDM.siteocc.iCAR

```r
# Computation time
Time.hSDM

# Outputs

# Parameter estimates
summary(mod.hSDM.siteocc.iCAR$mcmc)
pdf("Posterior_hSDM.siteocc.iCAR.pdf")
plot(mod.hSDM.siteocc.iCAR$mcmc)
dev.off()

# Predictions
summary(mod.hSDM.siteocc.iCAR$theta.latent)
summary(mod.hSDM.siteocc.iCAR$delta.latent)
summary(mod.hSDM.siteocc.iCAR$theta.pred)
pdf(file="Pred-Init.pdf")
plot(theta,mod.hSDM.siteocc.iCAR$theta.pred)
abline(a=0,b=1,col="red")
dev.off()

# Summary plots for spatial random effects

# rho.pred
rho.pred <- apply(mod.hSDM.siteocc.iCAR$rho.pred,2,mean)
r.rho.pred <- rasterFromXYZ(cbind(coords,rho.pred))

# plot
pdf(file="Summary_hSDM.siteocc.iCAR.pdf")
par(mfrow=c(2,2))
# rho target
plot(r.rho, main="rho target")
plot(sites.sp,add=TRUE)
# rho estimated
plot(r.rho.pred, main="rho estimated")
# correlation and "shrinkage"
Levels.cells <- sort(unique(cells))
plot(rho[-Levels.cells],rho.pred[-Levels.cells],
    xlim=range(rho),
    ylim=range(rho),
    xlab="rho target",
    ylab="rho estimated")
points(rho[Levels.cells],rho.pred[Levels.cells],pch=16,col="blue")
legend(x=-3,y=4,legend="Visited cells",col="blue",pch=16,bty="n")
abline(a=0,b=1,col="red")
dev.off()

## End(Not run)
```
hSDM.ZIB

**ZIB (Zero-Inflated Binomial) model**

**Description**

The hSDM.ZIB function can be used to model species distribution including different processes in a hierarchical Bayesian framework: a Bernoulli suitability process (refering to environmental suitability) and a Binomial observability process (refering to various ecological and methodological issues explaining the species detection). The hSDM.ZIB function calls a Gibbs sampler written in C code which uses a Metropolis algorithm to estimate the conditional posterior distribution of hierarchical model's parameters.

**Usage**

```r
hSDM.ZIB(presences, trials, suitability, observability, data, suitability.pred=NULL, burnin = 5000, mcmc = 10000, thin = 10, beta.start, gamma.start, mubeta = 0, Vbeta = 1e+06, mugamma = 0, Vgamma = 1e+06, seed = 1234, verbose = 1, save.p = 0)
```

**Arguments**

- **presences**: A vector indicating the number of successes (or presences) for each observation.
- **trials**: A vector indicating the number of trials for each observation. \( t_i \) should be superior to zero and superior or equal to \( y_i \), the number of successes for observation \( i \).
- **suitability**: A one-sided formula of the form \( \sim x_1 + \ldots + x_p \) with \( p \) terms specifying the explicative variables for the suitability process.
- **observability**: A one-sided formula of the form \( \sim w_1 + \ldots + w_q \) with \( q \) terms specifying the explicative variables for the observability process.
- **data**: A data frame containing the model’s variables.
- **suitability.pred**: An optional data frame in which to look for variables with which to predict. If NULL, the observations are used.
- **burnin**: The number of burnin iterations for the sampler.
- **mcmc**: The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to \( \text{burnin} + \text{mcmc} \). \( \text{burnin} + \text{mcmc} \) must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.
- **thin**: The thinning interval used in the simulation. The number of mcmc iterations must be divisible by this value.
- **beta.start**: Starting values for \( \beta \) parameters of the suitability process. This can either be a scalar or a \( p \)-length vector.
- **gamma.start**: Starting values for \( \beta \) parameters of the observability process. This can either be a scalar or a \( q \)-length vector.
mubeta  Means of the priors for the $\beta$ parameters of the suitability process. mubeta must be either a scalar or a p-length vector. If mubeta takes a scalar value, then that value will serve as the prior mean for all of the betas. The default value is set to 0 for an uninformative prior.

vbeta  Variances of the Normal priors for the $\beta$ parameters of the suitability process. vbeta must be either a scalar or a p-length vector. If vbeta takes a scalar value, then that value will serve as the prior variance for all of the betas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

mugamma  Means of the Normal priors for the $\gamma$ parameters of the observability process. mugamma must be either a scalar or a p-length vector. If mugamma takes a scalar value, then that value will serve as the prior mean for all of the gammas. The default value is set to 0 for an uninformative prior.

vgamma  Variances of the Normal priors for the $\gamma$ parameters of the observability process. vgamma must be either a scalar or a p-length vector. If vgamma takes a scalar value, then that value will serve as the prior variance for all of the gammas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

seed  The seed for the random number generator. Default set to 1234.

verbose  A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

save.p  A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the prob.p.pred vector. Be careful, setting save.p to 1 might require a large amount of memory.

Details

The model integrates two processes, an ecological process associated to the presence or absence of the species due to habitat suitability and an observation process that takes into account the fact that the probability of detection of the species is inferior to one.

Ecological process:

$$z_i \sim \text{Bernoulli}(\theta_i)$$
$$\logit(\theta_i) = X_i \beta$$

Observation process:

$$y_i \sim \text{Binomial}(z_i \ast \delta_i, t_i)$$
$$\logit(\delta_i) = W_i \gamma$$

Value

mcmc  An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance $D$, with $D = -2 \log(\prod_i P(y_i, z_i | \ldots))$, is also provided.

prob.p.pred  If save.p is set to 0 (default), prob.p.pred is the predictive posterior mean of the probability associated to the suitability process for each prediction. If save.p is set to 1, prob.p.pred is an mcmc object with sampled values of the probability associated to the suitability process for each prediction.
prob.p.latent  Predictive posterior mean of the probability associated to the suitability process for each observation.
prob.q.latent  Predictive posterior mean of the probability associated to the observability process for each observation.

Author(s)
Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>

References

See Also
plot.mcmc, summary.mcmc

Examples

```r
## Not run:

#-----------------------------------------------
# hSDM.ZIB()
# Example with simulated data
#-----------------------------------------------

#-------
#== Preambule
library(hSDM)

#----------------------
#== Data simulation

# Set seed for repeatability
seed <- 1234

# Number of observations
nobs <- 1000

# Target parameters
beta.target <- matrix(c(0.2, 0.5, 0.5), ncol=1)
gamma.target <- matrix(c(1), ncol=1)
```
## Uncomment if you want covariates on the observability process
## gamma.target <- matrix(c(0.2,0.5,0.5),nrow=1)

# Covariates for "suitability" process
set.seed(seed)
X1 <- rnorm(n=nobs,0,1)
set.seed(2*seed)
X2 <- rnorm(n=nobs,0,1)
X <- cbind(rep(1,nobs),X1,X2)

# Covariates for "observability" process
W <- cbind(rep(1,nobs))
## Uncomment if you want covariates on the observability process
## set.seed(3*seed)
## W1 <- rnorm(n=nobs,0,1)
## set.seed(4*seed)
## W2 <- rnorm(n=nobs,0,1)
## W <- cbind(rep(1,nobs),W1,W2)

### Simulating latent variables

# Suitability
logit.theta.1 <- X %*% beta.target
theta.1 <- inv.logit(logit.theta.1)
set.seed(seed)
y.1 <- rbinom(nobs,1,theta.1)

# Observability
set.seed(seed)
trials <- rpois(nobs,5) # Number of trial associated to each observation
trials[trials==0] <- 1
logit.theta.2 <- W %*% gamma.target
theta.2 <- inv.logit(logit.theta.2)
set.seed(seed)
y.2 <- rbinom(nobs,trials,theta.2)

### Simulating response variable
Y <- y.2*y.1

### Data-set
Data <- data.frame(Y,trials,X1,X2)
## Uncomment if you want covariates on the observability process
## Data <- data.frame(Y,trials,X1,X2,W1,W2)

### ZIB model
mod.hSDM.ZIB <- hSDM.ZIB(presences=Data$Y,
                         trials=Data$trials,
                         suitability=~X1+X2,
                         observability=~1, #=~1+W1+W2 if covariates
                         data=Data,
                         suitability.pred=NULL,
The `hsdm.ZIB.iCAR` function can be used to model species distribution including different processes in a hierarchical Bayesian framework: a Bernoulli suitability process (referring to environmental suitability) which takes into account the spatial dependence of the observations, and a Binomial observability process (referring to various ecological and methodological issues explaining the species detection). The `hsdm.ZIB.iCAR` function calls a Gibbs sampler written in C code which uses an adaptive Metropolis algorithm to estimate the conditional posterior distribution of hierarchical model's parameters.

Usage

```r
hsdm.ZIB.iCAR(presences, trials, suitability,
              observability, spatial.entity, data, n.neighbors, neighbors,
              suitability.pred=NULL, spatial.entity.pred=NULL, burnin = 5000, mcmc =
              10000, thin = 10, beta.start, gamma.start, Vrho.start, mubeta = 0, Vbeta
              = 1e+06, mugamma = 0, Vgamma = 1e+06, priorVrho = "1/Gamma", shape =
              0.5, rate = 0.0005, Vrho.max=1000, seed = 1234, verbose = 1, save.rho =
              0, save.p = 0)
```

Arguments

- `presences`: A vector indicating the number of successes (or presences) for each observation.
trials A vector indicating the number of trials for each observation. \( t_i \) should be superior to zero and superior or equal to \( y_i \), the number of successes for observation \( i \).

suitability A one-sided formula of the form \( \sim x_1 + ... + x_p \) with \( p \) terms specifying the explicative variables for the suitability process.

observability A one-sided formula of the form \( \sim w_1 + ... + w_q \) with \( q \) terms specifying the explicative variables for the observability process.

spatial.entity A vector indicating the spatial entity identifier (from one to the total number of entities) for each observation. Several observations can occur in one spatial entity. A spatial entity can be a raster cell for example.

data A data frame containing the model’s variables.

n.neighbors A vector of integers that indicates the number of neighbors (adjacent entities) of each spatial entity. length(n.neighbors) indicates the total number of spatial entities.

neighbors A vector of integers indicating the neighbors (adjacent entities) of each spatial entity. Must be of the form c(neighbors of entity 1, neighbors of entity 2, ..., neighbors of the last entity). Length of the neighbors vector should be equal to sum(n.neighbors).

suitability.pred An optional data frame in which to look for variables with which to predict. If NULL, the observations are used.

spatial.entity.pred An optional vector indicating the spatial entity identifier (from one to the total number of entities) for predictions. If NULL, the vector spatial.entity for observations is used.

burnin The number of burnin iterations for the sampler.

mcmc The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to burnin+mcmc. burnin+mcmc must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.

thin The thinning interval used in the simulation. The number of mcmc iterations must be divisible by this value.

beta.start Starting values for \( \beta \) parameters of the suitability process. This can either be a scalar or a \( p \)-length vector.

gamma.start Starting values for \( \beta \) parameters of the observability process. This can either be a scalar or a \( q \)-length vector.

Vrho.start Positive scalar indicating the starting value for the variance of the spatial random effects.

mubeta Means of the priors for the \( \beta \) parameters of the suitability process. mubeta must be either a scalar or a \( p \)-length vector. If mubeta takes a scalar value, then that value will serve as the prior mean for all of the betas. The default value is set to 0 for an uninformative prior.

Vbeta Variances of the Normal priors for the \( \beta \) parameters of the suitability process. Vbeta must be either a scalar or a \( p \)-length vector. If Vbeta takes a scalar value, then that value will serve as the prior variance for all of the betas. The default variance is large and set to 1.0E6 for an uninformative flat prior.
mugamma: Means of the Normal priors for the $\gamma$ parameters of the observability process. mugamma must be either a scalar or a p-length vector. If mugamma takes a scalar value, then that value will serve as the prior mean for all of the gammas. The default value is set to 0 for an uninformative prior.

Vgamma: Variances of the Normal priors for the $\gamma$ parameters of the observability process. Vgamma must be either a scalar or a p-length vector. If Vgamma takes a scalar value, then that value will serve as the prior variance for all of the gammas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

priorVrho: Type of prior for the variance of the spatial random effects. Can be set to a fixed positive scalar, or to an inverse-gamma distribution ("1/Gamma") with parameters shape and rate, or to a uniform distribution ("Uniform") on the interval [0,Vrho.max]. Default set to "1/Gamma".

shape: The shape parameter for the Gamma prior on the precision of the spatial random effects. Default value is shape=0.05 for uninformative prior.

rate: The rate (1/scale) parameter for the Gamma prior on the precision of the spatial random effects. Default value is rate=0.0005 for uninformative prior.

Vrho.max: Upper bound for the uniform prior of the spatial random effect variance. Default set to 1000.

seed: The seed for the random number generator. Default set to 1234.

verbose: A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

save.rho: A switch (0,1) which determines whether or not the sampled values for rhos are saved. Default is 0: the posterior mean is computed and returned in the rho.pred vector. Be careful, setting save.rho to 1 might require a large amount of memory.

save.p: A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the prob.p.pred vector. Be careful, setting save.p to 1 might require a large amount of memory.

Details

The model integrates two processes, an ecological process associated to the presence or absence of the species due to habitat suitability and an observation process that takes into account the fact that the probability of detection of the species is inferior to one. The ecological process includes an intrinsic conditional autoregressive model (iCAR) model for spatial autocorrelation between observations, assuming that the probability of presence of the species at one site depends on the probability of presence of the species on neighboring sites.

**Ecological process:**

$$z_i \sim Bernoulli(\theta_i)$$

$$\text{logit}(\theta_i) = X_i \beta + \rho_{j(i)}$$

$\rho_j$: spatial random effect

$j(i)$: index of the spatial entity for observation $i$. 
**Spatial autocorrelation:**
An intrinsic conditional autoregressive model (iCAR) is assumed:

$$\rho_j \sim \text{Normal}(\mu_j, V_\rho/n_j)$$

$\mu_j$: mean of $\rho_j$ in the neighborhood of $j$.
$V_\rho$: variance of the spatial random effects.
$n_j$: number of neighbors for spatial entity $j$.

**Observation process:**

$$y_i \sim \text{Binomial}(z_i \ast \delta_i, t_i)$$

$$\logit(\delta_i) = W_i \gamma$$

**Value**

- **mcmc**: An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance $D$, with $D = -2 \log(\prod_i P(y_i, z_i | ...))$, is also provided.

- **rho.pred**: If `save.rho` is set to 0 (default), `rho.pred` is the predictive posterior mean of the spatial random effect associated to each spatial entity. If `save.rho` is set to 1, `rho.pred` is an mcmc object with sampled values for each spatial random effect associated to each spatial entity.

- **prob.p.pred**: If `save.p` is set to 0 (default), `prob.p.pred` is the predictive posterior mean of the probability associated to the suitability process for each prediction. If `save.p` is set to 1, `prob.p.pred` is an mcmc object with sampled values of the probability associated to the suitability process for each prediction.

- **prob.q.latent**: Predictive posterior mean of the probability associated to the suitability process for each observation.

- **prob.q.latent**: Predictive posterior mean of the probability associated to the observability process for each observation.

**Author(s)**

Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>

**References**


See Also

plot.mcmc, summary.mcmc

Examples

## Not run:

```
# hSDM.ZIB.iCAR()
# Example with simulated data

#== Preambule
library(hSDM)
library(raster)
library(sp)
library(mvtnorm)

#== Data simulation
# Set seed for repeatability
seed <- 1234

# Target parameters
beta.target <- matrix(c(0.2,0.5,0.5),ncol=1)
gamma.target <- matrix(c(1),ncol=1)
# Uncomment if you want covariates on the observability process
## gamma.target <- matrix(c(0.2,0.5,0.5),ncol=1)
vrho.target <- 1 # Spatial Variance

# Landscape
Landscape <- raster(ncol=20,nrow=20,crs='+proj=utm +zone=1')
ncell <- ncell(Landscape)

# Neighbors
neighbors.mat <- adjacent(Landscape, cells=c(1:ncell), directions=8, pairs=TRUE, sorted=TRUE)
n.neighbors <- as.data.frame(table(as.factor(neighbors.mat[,1])))[,2]
adj <- neighbors.mat[,2]

# Generate symmetric adjacency matrix, A
A <- matrix(0,ncell,ncell)
index.start <- 1
for (i in 1:ncell) {
  index.end <- index.start+n.neighbors[i]-1
  A[1,adj[c(index.start:index.end)]] <- 1
  index.start <- index.end+1
}

# Spatial effects
d <- 1 # Spatial dependence parameter = 1 for intrinsic CAR
Q <- diag(n.neighbors)-d*A + diag(0.0001,n.cell) # Add small constant to make Q non-singular
covrho <- Vrho.target*solve(Q) # Covariance of rhos
set.seed(seed)
 rho <- c(rmvnorm(1,sigma= covrho)) # Spatial Random Effects
 rho <- rho-mean(rho) # Centering rhos on zero

# Visited cells
n.visited <- 150 # Compare with 400, 350 and 100 for example
set.seed(seed)
visited.cells <- sort(sample(1:n.cell,n.visited,replace=FALSE)) # Draw visited cells at random
notvisited.cells <- c(1:n.cell)[-visited.cells]

# Number of observations
n.obs <- 300

# Cell vector
set.seed(seed)
cells <- c(visited.cells,sample(visited.cells,n.obs-n.visited,replace=TRUE))
coords <- xyFromCell(Landscape,cells) # Get coordinates

# Covariates for "suitability" process
set.seed(seed)
X1.cell <- rnorm(n=n.cell,0,1)
set.seed(2*seed)
X2.cell <- rnorm(n=n.cell,0,1)
X1 <- X1.cell[cells]
X2 <- X2.cell[cells]
X <- cbind(rep(1,n.obs),X1,X2)

# Covariates for "observability" process
W <- cbind(rep(1,n.obs))
## Uncomment if you want covariates on the observability process
## set.seed(3*seed)
## W1 <- rnorm(n=n.obs,0,1)
## set.seed(4*seed)
## W2 <- rnorm(n=n.obs,0,1)
## W <- cbind(rep(1,n.obs),W1,W2)

#== Simulating latent variables

# Suitability
logit.theta.1 <- vector()
for (n in 1:n.obs) {
  logit.theta.1[n] <- X[n,]*%*%beta.target+rho[cells[n]]
}
theta.1 <- inv.logit(logit.theta.1)
set.seed(seed)
y.1 <- rbinom(n.obs,1,theta.1)

# Observability
set.seed(seed)
trials <- rpois(n.obs,5) # Number of trial associated to each observation
```r
trials[trials==0] <- 1
logit.theta.2 <- W*gamma.target
teta.2 <- inv.logit(logit.theta.2)
set.seed(seed)
y.2 <- rbinom(nobs,trials,theta.2)

## Simulating response variable
Y <- y.2*y.1

## Data-set
Data <- data.frame(Y, trials, cells, X1, X2)
## Uncomment if you want covariates on the observability process
## Data <- data.frame(Y, trials, cells, X1, X2, W1, W2)
Data <- SpatialPointsDataFrame(coords=data$data, data=Data)
plot(Data)

## Data-set for predictions (suitability on each spatial cell)
Data.pred <- data.frame(X1=X1.cell, X2=X2.cell, cells=c(1:ncell))

## Site-occupancy model
mod.hSDM.ZIB.iCAR <- hSDM.ZIB.iCAR(presences=data$data$Y,
  trials=data$data$trials,
  suitability=-X1+X2,
  observability=1,
  spatial.entity=data$data$cells,
  data=Data,
  n.neighbors=n.neighbors,
  neighbors=adj,
  ## suitability.pred=NULL,
  ## spatial.entity.pred=NULL,
  suitability.pred=Data.pred,
  spatial.entity.pred=Data.pred$cells,
  burnin=5000, mcmc=5000, thin=5,
  beta.start=0,
  gamma.start=0,
  Vrho.start=10,
  priorVrhorho="1/Gamma",
  #priorVrho="Uniform",
  #priorVrho=10,
  mubeta=0, Vbeta=0.05E6,
  mugamma=0, Vgamma=0.05E6,
  shape=0.5, rate=0.0005,
  #Vrho.max=1000,
  seed=1234, verbose=1,
  save.rho=1, save.p=0)

## Outputs
## Parameter estimates
summary(mod.hSDM.ZIB.iCAR$mcmc)
```
hSDM.ZIB.iCAR.alteration

---

hSDM.ZIB.iCAR.alteration

ZIB (Zero-Inflated Binomial) model with CAR process taking into account site alteration
Description

The hSDM.ZIB.iCAR.alteration function can be used to model species distribution including different processes in a hierarchical Bayesian framework: (i) a Bernoulli suitability process (referring to environmental suitability) which takes into account the spatial dependence of the observations, (ii) an alteration process (referring to anthropogenic disturbances), and (iii) a Binomial observability process (referring to various ecological and methodological issues explaining the species detection). The hSDM.ZIB.iCAR.alteration function calls a Gibbs sampler written in C code which uses an adaptive Metropolis algorithm to estimate the conditional posterior distribution of hierarchical model’s parameters.

Usage

hSDM.ZIB.iCAR.alteration(presences, trials, suitability, observability, spatial.entity, alteration, data, n.neighbors, neighbors, suitability.pred=NULL, spatial.entity.pred=NULL, burnin = 5000, mcmc = 10000, thin = 10, beta.start, gamma.start, Vrho.start, mubeta = 0, Vbeta = 1e+06, mugamma = 0, Vgamma = 1e+06, priorVrho = "1/Gamma", shape = 0.5, rate = 0.0005, Vrho.max=1000, seed = 1234, verbose = 1, save.rho = 0, save.p = 0)

Arguments

- presences: A vector indicating the number of successes (or presences) for each observation.
- trials: A vector indicating the number of trials for each observation. \( t_i \) should be superior to zero and superior or equal to \( y_i \), the number of successes for observation \( i \).
- suitability: A one-sided formula of the form \( \sim x_1 + ... + x_p \) with \( p \) terms specifying the explicative variables for the suitability process.
- observability: A one-sided formula of the form \( \sim w_1 + ... + w_q \) with \( q \) terms specifying the explicative variables for the observability process.
- spatial.entity: A vector indicating the spatial entity identifier (from one to the total number of entities) for each observation. Several observations can occur in one spatial entity. A spatial entity can be a raster cell for example.
- alteration: A vector indicating the proportion of area in the spatial cell which is transformed (by anthropogenic activities for example) for each observation. Must be between 0 and 1.
- data: A data frame containing the model’s variables.
- n.neighbors: A vector of integers that indicates the number of neighbors (adjacent entities) of each spatial entity. \texttt{length(n.neighbors)} indicates the total number of spatial entities.
- neighbors: A vector of integers indicating the neighbors (adjacent entities) of each spatial entity. Must be of the form \texttt{c(neighbors of entity 1, neighbors of entity 2, ..., neighbors of the last entity)}. Length of the \texttt{neighbors} vector should be equal to \texttt{sum(n.neighbors)}.
- suitability.pred: An optional data frame in which to look for variables with which to predict. If NULL, the observations are used.
spatial.entity.pred
An optional vector indicating the spatial entity identifier (from one to the total number of entities) for predictions. If NULL, the vector spatial.entity for observations is used.

burnin
The number of burnin iterations for the sampler.

mcmc
The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to burnin+mcmc. burnin+mcmc must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.

thin
The thinning interval used in the simulation. The number of mcmc iterations must be divisible by this value.

beta.start
Starting values for $\beta$ parameters of the suitability process. This can either be a scalar or a $p$-length vector.

gamma.start
Starting values for $\beta$ parameters of the observability process. This can either be a scalar or a $q$-length vector.

Vrho.start
Positive scalar indicating the starting value for the variance of the spatial random effects.

mubeta
Means of the priors for the $\beta$ parameters of the suitability process. mubeta must be either a scalar or a $p$-length vector. If mubeta takes a scalar value, then that value will serve as the prior mean for all of the betas. The default value is set to 0 for an uninformative prior.

Vbeta
Variances of the Normal priors for the $\beta$ parameters of the suitability process. Vbeta must be either a scalar or a $p$-length vector. If Vbeta takes a scalar value, then that value will serve as the prior variance for all of the betas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

mugamma
Means of the Normal priors for the $\gamma$ parameters of the observability process. mugamma must be either a scalar or a $q$-length vector. If mugamma takes a scalar value, then that value will serve as the prior mean for all of the gammas. The default value is set to 0 for an uninformative prior.

Vgamma
Variances of the Normal priors for the $\gamma$ parameters of the observability process. Vgamma must be either a scalar or a $q$-length vector. If Vgamma takes a scalar value, then that value will serve as the prior variance for all of the gammas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

priorVrho
Type of prior for the variance of the spatial random effects. Can be set to a fixed positive scalar, or to an inverse-gamma distribution ("1/Gamma") with parameters shape and rate, or to a uniform distribution ("Uniform") on the interval [0,Vrho.max]. Default set to "1/Gamma".

shape
The shape parameter for the Gamma prior on the precision of the spatial random effects. Default value is shape=0.05 for uninformative prior.

rate
The rate (1/scale) parameter for the Gamma prior on the precision of the spatial random effects. Default value is rate=0.0005 for uninformative prior.

Vrho.max
Upper bound for the uniform prior of the spatial random effect variance. Default set to 1000.

seed
The seed for the random number generator. Default set to 1234.
verbose
A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

save.rho
A switch (0,1) which determines whether or not the sampled values for rhos are saved. Default is 0: the posterior mean is computed and returned in the rho.pred vector. Be careful, setting save.rho to 1 might require a large amount of memory.

save.p
A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the prob.p.pred vector. Be careful, setting save.p to 1 might require a large amount of memory.

Details
The model integrates two processes, an ecological process associated to the presence or absence of the species due to habitat suitability and an observation process that takes into account the fact that the probability of detection of the species is inferior to one. The ecological process includes an intrinsic conditional autoregressive model (iCAR) model for spatial autocorrelation between observations, assuming that the probability of presence of the species at one site depends on the probability of presence of the species on neighboring sites.

Ecological process:
\[ z_i \sim Bernoulli(\theta_i) \]
\[ \text{logit}(\theta_i) = X_i \beta + \rho_j(i) \]
\[ \rho_j: \text{spatial random effect} \]
\[ j(i): \text{index of the spatial entity for observation } i. \]

Spatial autocorrelation:
An intrinsic conditional autoregressive model (iCAR) is assumed:
\[ \rho_j \sim \text{Normal}(\mu_j, V_\rho / n_j) \]
\[ \mu_j: \text{mean of } \rho_j, \text{ in the neighborhood of } j. \]
\[ V_\rho: \text{variance of the spatial random effects.} \]
\[ n_j: \text{number of neighbors for spatial entity } j. \]

Observation process:
\[ y_i \sim \text{Binomial}(z_i \ast \delta_i, t_i) \]
\[ \text{logit}(\delta_i) = W_i \gamma \]

Value
mcmc
An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance \( D \), with \( D = -2 \log(\prod_i P(y_i, z_i|\ldots)) \), is also provided.

rho.pred
If save.rho is set to 0 (default), rho.pred is the predictive posterior mean of the spatial random effect associated to each spatial entity. If save.rho is set to 1, rho.pred is an mcmc object with sampled values for each spatial random effect associated to each spatial entity.
prob.p.pred  If `save.p` is set to 0 (default), `prob.p.pred` is the predictive posterior mean of the probability associated to the suitability process for each prediction. If `save.p` is set to 1, `prob.p.pred` is an `mcmc` object with sampled values of the probability associated to the suitability process for each prediction.

prob.p.latent  Predictive posterior mean of the probability associated to the suitability process for each observation.

prob.q.latent  Predictive posterior mean of the probability associated to the observability process for each observation.

Author(s)

Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>

References


See Also

`plot.mcmc`, `summary.mcmc`

Examples

```r
## Not run:
#==============================================================
# hSDM.ZIB.iCAR.alteration()
# Example with simulated data
#==============================================================

#========
#== Preambule
library(hSDM)
library(raster)
library(sp)
library(mvtnorm)

#========
#== Data simulation
# Set seed for repeatability
seed <- 1234
```
# Target parameters
beta.target <- matrix(c(0.2,0.5,0.5),ncol=1)
gamma.target <- matrix(c(1),ncol=1)
## Uncomment if you want covariates on the observability process
## gamma.target <- matrix(c(0.2,0.5,0.5),ncol=1)
Vrho.target <- 1 # Spatial Variance

# Landscape
Landscape <- raster(ncol=20,nrow=20,crs='+proj=utm +zone=1')
ncell <- ncell(Landscape)

# Neighbors
neighbors.mat <- adjacent(Landscape, cells=1:ncell, directions=8, pairs=TRUE, sorted=TRUE)
n.neighbors <- as.data.frame(table(as.factor(neighbors.mat[,1])))[,2]
adj <- neighbors.mat[,2]

# Generate symmetric adjacency matrix, A
A <- matrix(0,ncell,ncell)
index.start <- 1
for (i in 1:ncell) {
  index.end <- index.start+n.neighbors[i]-1
  A[i,adj[c(index.start:index.end)]] <- 1
  index.start <- index.end+1
}

# Spatial effects
d <- 1 # Spatial dependence parameter = 1 for intrinsic CAR
Q <- diag(n.neighbors)-d*A + diag(.0001,ncell) # Add small constant to make Q non-singular
covrho <- Vrho.target*solve(Q) # Covariance of rhos
set.seed(seed)
rho <- rmvnorm(1,sigma=covrho) # Spatial Random Effects
rho <- rho-mean(rho) # Centering rhos on zero

# Visited cells
n.visited <- 150 # Compare with 400, 350 and 100 for example
set.seed(seed)
visited.cells <- sort(sample(1:ncell,n.visited,replace=FALSE)) # Draw visited cells at random
notvisited.cells <- c(1:ncell)[-visited.cells]

# Number of observations
nobs <- 300

# Cell vector
set.seed(seed)
cells <- c(visited.cells,sample(visited.cells,notvisited.cells,nobs-n.visited,replace=TRUE))
coords <- xyFromCell(Landscape,cells) # Get coordinates

# Covariates for "suitability" process
set.seed(seed)
X1.cell <- rnorm(n=ncell,0,1)
set.seed(2*seed)
X2.cell <- rnorm(n=ncell,0,1)
X1 <- X1.cell[cells]
X2 <- X2.cell[cells]
X <- cbind(rep(1,nobs),X1,X2)

# Alteration
U <- runif(n=nobs,min=0,max=1)

# Covariates for "observability" process
W <- cbind(rep(1,nobs))
## Uncomment if you want covariates on the observability process
## set.seed(3*seed)
## W1 <- rnorm(n=nobs,0,1)
## set.seed(4*seed)
## W2 <- rnorm(n=nobs,0,1)
## W <- cbind(rep(1,nobs),W1,W2)

### Simulating latent variables

### Suitability
logit.theta.1 <- vector()
for (n in 1:nobs) {
  logit.theta.1[n] <- X[n,]*%beta.target+rho[cells[n]]
}
theta.1 <- inv.logit(logit.theta.1)
set.seed(seed)
y.1 <- rbinom(nobs,1,theta.1)

### Observability
u <- rbinom(nobs,1,U)

### Data-set
Data <- data.frame(Y, trials, cells, X1, X2, U)
## Uncomment if you want covariates on the observability process
## Data <- data.frame(Y, trials, cells, X1, X2, W1, W2, U)
Data <- SpatialPointsDataFrame(coords=coords, data=Data)
plot(Data)

### Data-set for predictions (suitability on each spatial cell)
Data.pred <- data.frame(X1=X1.cell, X2=X2.cell, cells=c(1:ncell))

### Data-set for predictions (suitability on each spatial cell)
Data.pred <- data.frame(X1=X1.cell, X2=X2.cell, cells=c(1:ncell))
### Site-occupancy model

```r
mod.hSDM.ZIB.iCAR.alteration <- hSDM.ZIB.iCAR.alteration(
presences=Data$Y,
  trials=Data$trials,
  suitability=-X1+X2,
  observability=1,
  spatial.entity=Data$cells,
  alteration=Data$U,
  data=Data,
  n.neighbors=n.neighbors,
  neighbors=adj,
  ## suitability.pred=NULL,
  ## spatial.entity.pred=NULL,
  suitability.pred=Data.pred,
  spatial.entity.pred=Data.pred$cells,
  burnin=5000, mcmc=5000, thin=5,
  beta.start=0,
  gamma.start=0,
  Vrho.start=10,
  priorVrho="1/Gamma",
  #priorVrho="Uniform",
  #priorVrho=10,
  mubeta=0, Vbeta=1.0E6,
  mugamma=0, Vgamma=1.0E6,
  shape=0.5, rate=0.0005,
  #Vrho.max=1000,
  seed=1234, verbose=1,
  save.rho=1, save.p=0)
```

### Outputs

#### Parameter estimates

```r
summary(mod.hSDM.ZIB.iCAR.alteration$mcmc)
```

#### MCMC and posteriors

```r
df <- read.csv("Posteriors.hSDM.ZIB.iCAR.alteration.csv")
plot(mod.hSDM.ZIB.iCAR.alteration$mcmc)
dev.off()

df <- read.csv("Posteriors.rho.hSDM.ZIB.iCAR.alteration.csv")
plot(mod.hSDM.ZIB.iCAR.alteration$rho.pred)
dev.off()
```

#### Summary plots

```r
# rho
r.rho <- r.rho.pred <- r.visited <- Landscape
r.rho[] <- rho
r.rho.pred <- apply(mod.hSDM.ZIB.iCAR.alteration$rho.pred,2,mean)
r.rho.pred[] <- r.rho.pred
r.visited[] <- 0
r.visited[visited.cells] <- 1
```
hSDM.ZIP

ZIP (Zero-Inflated Poisson) model

Description

The hSDM.ZIP function can be used to model species distribution including different processes in a hierarchical Bayesian framework: a Bernoulli suitability process (referring to various ecological variables explaining environmental suitability or not) and a Poisson abundance process (referring to various ecological variables explaining the species abundance when the habitat is suitable). The hSDM.ZIP function calls a Gibbs sampler written in C code which uses a Metropolis algorithm to estimate the conditional posterior distribution of hierarchical model's parameters.

Usage

hSDM.ZIP(counts, suitability, abundance, data,
suitability.pred=NULL, burnin = 5000, mcmc = 10000, thin = 10,
beta.start, gamma.start, mubeta = 0, Vbeta = 1e+06, mugamma = 0, Vgamma = 1e+06, seed = 1234, verbose = 1, save.p = 0)

Arguments

counts A vector indicating the count for each observation.
suitability A one-sided formula of the form $x_1 + ... + x_p$ with $p$ terms specifying the explicative variables for the suitability process.
abundance

A one-sided formula of the form $\sim w_1 + ... + w_q$ with $q$ terms specifying the explicative variables for the abundance process.

data

A data frame containing the model’s variables.

suitability.pred

An optional data frame in which to look for variables with which to predict. If NULL, the observations are used.

burnin

The number of burnin iterations for the sampler.

mcmc

The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to burnin+mcmc. burnin+mcmc must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.

thin

The thinning interval used in the simulation. The number of mcmc iterations must be divisible by this value.

beta.start

Starting values for $\beta$ parameters of the suitability process. This can either be a scalar or a $p$-length vector.

gamma.start

Starting values for $\beta$ parameters of the abundance process. This can either be a scalar or a $q$-length vector.

mubeta

Means of the priors for the $\beta$ parameters of the suitability process. $mubeta$ must be either a scalar or a $p$-length vector. If $mubeta$ takes a scalar value, then that value will serve as the prior mean for all of the betas. The default value is set to 0 for an uninformative prior.

Vbeta

Variances of the Normal priors for the $\beta$ parameters of the suitability process. $Vbeta$ must be either a scalar or a $p$-length vector. If $Vbeta$ takes a scalar value, then that value will serve as the prior variance for all of the betas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

mugamma

Means of the Normal priors for the $\gamma$ parameters of the abundance process. $mugamma$ must be either a scalar or a $p$-length vector. If $mugamma$ takes a scalar value, then that value will serve as the prior mean for all of the gammas. The default value is set to 0 for an uninformative prior.

Vgamma

Variances of the Normal priors for the $\gamma$ parameters of the abundance process. $Vgamma$ must be either a scalar or a $p$-length vector. If $Vgamma$ takes a scalar value, then that value will serve as the prior variance for all of the gammas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

seed

The seed for the random number generator. Default set to 1234.

verbose

A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

save.p

A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the prob.p.pred vector. Be careful, setting save.p to 1 might require a large amount of memory.

Details

The model integrates two processes, an ecological process associated to habitat suitability (habitat is suitable or not for the species) and an abundance process that takes into account ecological variables explaining the species abundance when the habitat is suitable.
Suitability process:

\[ z_i \sim \text{Bernoulli}(\theta_i) \]
\[ \logit(\theta_i) = X_i\beta \]

Abundance process:

\[ y_i \sim \text{Poisson}(z_i \ast \lambda_i) \]
\[ \log(\lambda_i) = W_i\gamma \]

Value

- **mcmc**: An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance \( D \), with \( D = -2 \log(\prod P(y_i, z_i|...)) \), is also provided.
- **prob.p.pred**: If `save.p` is set to 0 (default), `prob.p.pred` is the predictive posterior mean of the probability associated to the suitability process for each prediction. If `save.p` is set to 1, `prob.p.pred` is an mcmc object with sampled values of the probability associated to the suitability process for each prediction.
- **prob.p.latent**: Predictive posterior mean of the probability associated to the suitability process for each observation.
- **prob.q.latent**: Predictive posterior mean of the probability associated to the abundance process for each observation.

Author(s)

Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>

References


See Also

- `plot.mcmc`, `summary.mcmc`

Examples

```r
## Not run:

#----------------------------
# hSDM.ZIP()
# Example with simulated data```
# Preambulle
library(hSDM)

# Data simulation
# Set seed for repeatability
seed <- 1234

# Number of observations
nobs <- 1000

# Target parameters
beta.target <- matrix(c(0.2,0.5,0.5),ncol=1)
gamma.target <- matrix(c(1),ncol=1)

# Uncomment if you want covariates on the abundance process
## gamma.target <- matrix(c(0.2,0.5,0.5),ncol=1)

# Covariates for "suitability" process
set.seed(seed)
X1 <- rnorm(n=nobs,0,1)
set.seed(2*seed)
X2 <- rnorm(n=nobs,0,1)
X <- cbind(rep(1,nobs),X1,X2)

# Covariates for "abundance" process
W <- cbind(rep(1,nobs))

## Uncomment if you want covariates on the abundance process
## set.seed(3*seed)
## W1 <- rnorm(n=nobs,0,1)
## set.seed(4*seed)
## W2 <- rnorm(n=nobs,0,1)
## W <- cbind(rep(1,nobs),W1,W2)

### Simulating latent variables

# Suitability
logit.theta <- X %*% beta.target
theta <- inv.logit(logit.theta)
set.seed(seed)
y.1 <- rbinom(nobs,1,theta)

# Abundance
set.seed(seed)
log.lambda <- W %*% gamma.target
lambda <- exp(log.lambda)
set.seed(seed)
y.2 <- rpois(nobs,lambda)

### Simulating response variable
Y <- y.2*y.1

### Data-set
Data <- data.frame(Y,X1,X2)
## Uncomment if you want covariates on the abundance process
## Data <- data.frame(Y,X1,X2,W1,W2)

#-------------------------------
### ZIP model
mod.hSDM.ZIP <- hSDM.ZIP(counts=Data$Y,
suitability=-X1+X2,
abundance=-1, #=-1+W1+2 if covariates
data=Data,
suitability.pred=NULL,
burnin=1000, mcmc=1000, thin=5,
beta.start=0,
gamma.start=0,
mubeta=0, Vbeta=1.0E6,
mugamma=0, Vgamma=1.0E6,
seed=1234, verbose=1,
save.p=0)

#----------
### Outputs
pdf(file="Posterior.hSDM.ZIP.pdf")
plot(mod.hSDM.ZIP$mcmc)
dev.off()
summary(mod.hSDM.ZIP$prob.p.latent)
summary(mod.hSDM.ZIP$prob.q.latent)
summary(mod.hSDM.ZIP$prob.p.pred)

## End(Not run)

---

**hSDM.ZIP.iCAR**  
**ZIP (Zero-Inflated Poisson) model with CAR process**

**Description**

The hSDM.ZIP.iCAR function can be used to model species distribution including different processes in a hierarchical Bayesian framework: a *Bernoulli* suitability process (referring to various ecological variables explaining environmental suitability or not) which takes into account the spatial dependence of the observations, and a *Poisson* abundance process (referring to various ecological variables explaining the species abundance when the habitat is suitable). The hSDM.ZIP.iCAR function calls a Gibbs sampler written in C code which uses an adaptive Metropolis algorithm to estimate the conditional posterior distribution of hierarchical model’s parameters.
Usage

hSDM.ZIP.iCAR(counts, suitability, abundance, spatial.entity, data, n.neighbors, neighbors, suitability.pred=NULL, spatial.entity.pred=NULL, burnin = 5000, mcmc = 10000, thin = 10, beta.start, gamma.start, Vrho.start, mubeta = 0, Vbeta = 1e+06, mugamma = 0, Vgamma = 1e+06, priorVrho = "1/Gamma", shape = 0.5, rate = 0.0005, Vrho.max=1000, seed = 1234, verbose = 1, save.rho = 0, save.p = 0)

Arguments

counts A vector indicating the count for each observation.
suitability A one-sided formula of the form ∼ x_1 + ... + x_p with p terms specifying the explicative variables for the suitability process.
abundance A one-sided formula of the form ∼ w_1 + ... + w_q with q terms specifying the explicative variables for the abundance process.
spatial.entity A vector indicating the spatial entity identifier (from one to the total number of entities) for each observation. Several observations can occur in one spatial entity. A spatial entity can be a raster cell for example.
data A data frame containing the model's variables.
n.neighbors A vector of integers that indicates the number of neighbors (adjacent entities) of each spatial entity. length(n.neighbors) indicates the total number of spatial entities.
neighbors A vector of integers indicating the neighbors (adjacent entities) of each spatial entity. Must be of the form c(neighbors of entity 1, neighbors of entity 2, ... , neighbors of the last entity). Length of the neighbors vector should be equal to sum(n.neighbors).
suitability.pred An optional data frame in which to look for variables with which to predict. If NULL, the observations are used.
spatial.entity.pred An optional vector indicating the spatial entity identifier (from one to the total number of entities) for predictions. If NULL, the vector spatial.entity for observations is used.
burnin The number of burnin iterations for the sampler.
mcmc The number of Gibbs iterations for the sampler. Total number of Gibbs iterations is equal to burnin+mcmc. burnin+mcmc must be divisible by 10 and superior or equal to 100 so that the progress bar can be displayed.
thin The thinning interval used in the simulation. The number of mcmc iterations must be divisible by this value.
beta.start Starting values for β parameters of the suitability process. This can either be a scalar or a p-length vector.
gamma.start Starting values for β parameters of the observability process. This can either be a scalar or a q-length vector.
Vrho.start Positive scalar indicating the starting value for the variance of the spatial random effects.
mubeta  Means of the priors for the $\beta$ parameters of the suitability process. mubeta must be either a scalar or a p-length vector. If mubeta takes a scalar value, then that value will serve as the prior mean for all of the betas. The default value is set to 0 for an uninformative prior.

Vbeta  Varpriors of the Normal priors for the $\beta$ parameters of the suitability process. Vbeta must be either a scalar or a p-length vector. If Vbeta takes a scalar value, then that value will serve as the prior variance for all of the betas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

mugamma  Means of the Normal priors for the $\gamma$ parameters of the observability process. mugamma must be either a scalar or a p-length vector. If mugamma takes a scalar value, then that value will serve as the prior mean for all of the gammas. The default value is set to 0 for an uninformative prior.

vgamma  Variances of the Normal priors for the $\gamma$ parameters of the observability process. vgamma must be either a scalar or a p-length vector. If vgamma takes a scalar value, then that value will serve as the prior variance for all of the gammas. The default variance is large and set to 1.0E6 for an uninformative flat prior.

priorVrho  Type of prior for the variance of the spatial random effects. Can be set to a fixed positive scalar, or to an inverse-gamma distribution ("1/Gamma") with parameters shape and rate, or to a uniform distribution ("Uniform") on the interval [0,Vrho.max]. Default set to "1/Gamma".

shape  The shape parameter for the Gamma prior on the precision of the spatial random effects. Default value is shape=0.05 for uninformative prior.

rate  The rate (1/scale) parameter for the Gamma prior on the precision of the spatial random effects. Default value is rate=0.0005 for uninformative prior.

Vrho.max  Upper bound for the uniform prior of the spatial random effect variance. Default set to 1000.

seed  The seed for the random number generator. Default set to 1234.

verbose  A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

save.rho  A switch (0,1) which determines whether or not the sampled values for rhos are saved. Default is 0: the posterior mean is computed and returned in the rho.pred vector. Be careful, setting save.rho to 1 might require a large amount of memory.

save.p  A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the prob.p.pred vector. Be careful, setting save.p to 1 might require a large amount of memory.

Details

The model integrates two processes, an ecological process associated to habitat suitability (habitat is suitable or not for the species) and an abundance process that takes into account ecological variables explaining the species abundance when the habitat is suitable. The suitability process includes an intrinsic conditional autoregressive model (iCAR) model for spatial autocorrelation between
observations, assuming that the suitability at one site depends on the suitability on neighboring sites.

**Suitability process:**

\[ z_i \sim Bernoulli(\theta_i) \]

\[ \logit(\theta_i) = X_i \beta + \rho_j(i) \]

\( \rho_j \): spatial random effect

\( j(i) \): index of the spatial entity for observation \( i \).

**Spatial autocorrelation:**

An intrinsic conditional autoregressive model (iCAR) is assumed:

\[ \rho_j \sim Normal(\mu_j, V_{\rho}/n_j) \]

\( \mu_j \): mean of \( \rho_j \) in the neighborhood of \( j \).

\( V_{\rho} \): variance of the spatial random effects.

\( n_j \): number of neighbors for spatial entity \( j \).

**Abundance process:**

\[ y_i \sim Poisson(z_i * \lambda_i) \]

\[ \log(\lambda_i) = W_i \gamma \]

**Value**

- **mcmc**: An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance \( D \), with \( D = -2 \log(\prod_i P(y_i, z_i|...)) \), is also provided.

- **rho.pred**: If `save.rho` is set to 0 (default), `rho.pred` is the predictive posterior mean of the spatial random effect associated to each spatial entity. If `save.rho` is set to 1, `rho.pred` is an mcmc object with sampled values for each spatial random effect associated to each spatial entity.

- **prob.p.pred**: If `save.p` is set to 0 (default), `prob.p.pred` is the predictive posterior mean of the probability associated to the suitability process for each prediction. If `save.p` is set to 1, `prob.p.pred` is an mcmc object with sampled values of the probability associated to the suitability process for each prediction.

- **prob.p.latent**: Predictive posterior mean of the probability associated to the suitability process for each observation.

- **prob.q.latent**: Predictive posterior mean of the probability associated to the observability process for each observation.

**Author(s)**

Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>
References


See Also

`plot.mcmc`, `summary.mcmc`

Examples

```r
## Not run:

#============================================
# hSDM.ZIP.iCAR()
# Example with simulated data
#============================================

#== Preambule
library(hSDM)
library(raster)
library(sp)
library(mvtnorm)

#== Data simulation

# Set seed for repeatability
seed <- 1234

# Target parameters
beta.target <- matrix(c(0.2,0.5,0.5),ncol=1)
gamma.target <- matrix(c(1),ncol=1)
## Uncomment if you want covariates on the observability process
## gamma.target <- matrix(c(0.2,0.5,0.5),ncol=1)
Vrho.target <- 1 # Spatial Variance

# Landscape
Landscape <- raster(ncol=20,nrow=20,crs='+proj=utm +zone=1')
n.cell <- ncell(Landscape)

# Neighbors
neighbors.mat <- adjacent(Landscape, cells=c(1:n.cell), directions=8, pairs=TRUE, sorted=TRUE)
n.neighbors <- as.data.frame(table(as.factor(neighbors.mat[,1])))[,2]
adj <- neighbors.mat[,2]
```
# Generate symmetric adjacency matrix, A
A <- matrix(0, ncell, ncell)
index.start <- 1
for (i in 1:ncell) {
    index.end <- index.start+n.neighbors[i]-1
    A[i, adj[c(index.start:index.end)]] <- 1
    index.start <- index.end+1
}

# Spatial effects
d <- 1 # Spatial dependence parameter = 1 for intrinsic CAR
Q <- diag(n.neighbors)-d*A + diag(0.0001,ncell) # Add small constant to make Q non-singular
covrho <- Vrho.target*solve(Q) # Covariance of rhos
set.seed(seed)
rho <- c(rmvnorm(1,sigma=covrho)) # Spatial Random Effects
rho <- rho-mean(rho) # Centering rhos on zero

# Visited cells
n.visited <- 150 # Compare with 400, 350 and 100 for example
set.seed(seed)
visited.cells <- sort(sample(1:ncell,n.visited,replace=FALSE)) # Draw visited cells at random
notvisited.cells <- c(1:ncell)[-visited.cells]

# Number of observations
nobs <- 300

# Cell vector
set.seed(seed)
cells <- c(visited.cells, sample(visited.cells, nobs-n.visited, replace=TRUE))
coords <- xyFromCell(Landscape, cells) # Get coordinates

# Covariates for "suitability" process
set.seed(seed)
X1.cell <- rnorm(n=ncell, 0, 1)
set.seed(2*seed)
X2.cell <- rnorm(n=ncell, 0, 1)
X1 <- X1.cell[cells]
X2 <- X2.cell[cells]
X <- cbind(rep(1, nobs), X1, X2)

# Covariates for "abundance" process
W <- cbind(rep(1, nobs))
## Uncomment if you want covariates on the observability process
## set.seed(3*seed)
## W1 <- rnorm(n=nobs, 0, 1)
## set.seed(4*seed)
## W2 <- rnorm(n=nobs, 0, 1)
## W <- cbind(rep(1, nobs), W1, W2)

# Simulating latent variables

# Suitability
logit.theta <- vector()
for (n in 1:nobs) {
    logit.theta[n] <- X[n,]*beta.target+rho[cells[n]]
}
theta <- inv.logit(logit.theta)
set.seed(seed)
y.1 <- rbinom(nobs,1,theta)

# Abundance
set.seed(seed)
log.lambda <- W*gamma.target
lambda <- exp(log.lambda)
set.seed(seed)
y.2 <- rpois(nobs,lambda)

###== Simulating response variable
Y <- y.2*y.1

###== Data-set
Data <- data.frame(Y,cells,X1,X2)
## Uncomment if you want covariates on the observability process
## Data <- data.frame(Y,cells,X1,X2,W1,W2)
Data <- SpatialPointsDataFrame(coords=coords, data=Data)
plot(Data)

###== Data-set for predictions (suitability on each spatial cell)
Data.pred <- data.frame(X1=X1.cell,X2=X2.cell,cells=c(1:ncell))

=================================
###== ZIP model with CAR

mod.hSDM.ZIP.iCAR <- hSDM.ZIP.iCAR(counts=Data$Y,
suitability=X1+X2,
abundance=1,
spatial.entity=Data$cells,
data=Data,
n.neighbors=n.neighbors,
neighbors=adj,
suitability.pred=Data.pred,
spatial.entity.pred=Data.pred$cells,
burnin=5000, mcmc=5000, thin=5,
beta.start=0,
gamma.start=0,
Vrho.start=10,
priorVrho="1/Gamma",
#priorVrho="Uniform",
#priorVrho=10,
mubeta=0, Vbeta=1.0E6,
mugamma=0, Vgamma=1.0E6,
shape=0.5, rate=0.0005,
Vrho.max=1000,
seed=1234, verbose=1,
save.rho=1, save.p=0)
## Outputs

### Parameter estimates

```r
summary(mod.hSDM.ZIP.iCAR$mcmc)
```

### MCMC and posteriors

```r
df=
plot(mod.hSDM.ZIP.iCAR$mcmc)
dev.off()

pdf(file="Posteriors.rho_hSDM.ZIP.iCAR.pdf")
plot(mod.hSDM.ZIP.iCAR$rho.pred)
dev.off()
```

### Summary plots

```r
# rho
r.rho <- r.rho.pred <- r.visited <- Landscape
r.rho[] <- rho
r.rho.pred <- apply(mod.hSDM.ZIP.iCAR$rho.pred,2,mean)

r.rho.pred[] <- rho.pred
r.visited[] <- 0
r.visited[visited.cells] <- tapply(Data[,cells,mean]
# prob.p
r.prob.p <- Landscape
r.prob.p[] <- mod.hSDM.ZIP.iCAR$prob.p.pred

pdf(file="Summary_hSDM.ZIP.iCAR.pdf")
par(mfrow=c(3,2))
plot(r.rho, main="rho target")
plot(r.visited, main="Visited cells and counts")
plot(Data, add=TRUE, pch=16, cex=0.5)
plot(r.rho.pred, main="rho estimated")

plot(rho[visited.cells], rho.pred[visited.cells],
 xlab="rho target",
ylab="rho estimated")
points(rho[notvisited.cells], rho.pred[notvisited.cells], pch=16, col="blue")
legend(x=-4, y=3.5, legend="Unvisited cells", col="blue", pch=16, bty="n")
abline(a=0, b=1, col="red")
plot(r.prob.p, main="Predicted counts")
plot(Data, add=TRUE, pch=16, cex=0.5)
dev.off()

## End (Not run)
hSDM.ZIP.iCAR.alteration

ZIP (Zero-Inflated Poisson) model with CAR process taking into account site alteration

Description

The hSDM.ZIP.iCAR.alteration function can be used to model species distribution including different processes in a hierarchical Bayesian framework: (i) a Bernoulli suitability process (referring to various ecological variables explaining environmental suitability or not) which takes into account the spatial dependence of the observations, (ii) an alteration process (referring to anthropogenic disturbances), and (iii) a Poisson abundance process (referring to various ecological variables explaining the species abundance when the habitat is suitable). The hSDM.ZIP.iCAR.alteration function calls a Gibbs sampler written in C code which uses an adaptive Metropolis algorithm to estimate the conditional posterior distribution of hierarchical model’s parameters.

Usage

hSDM.ZIP.iCAR.alteration(counts, suitability, abundance, spatial.entity, alteration, data, n.neighbors, neighbors, suitability.pred=NULL, spatial.entity.pred=NULL, burnin = 5000, mcmc = 10000, thin = 10, beta.start, gamma.start, Vrho.start, mubeta = 0, Vbeta = 1e+06, mugamma = 0, Vgamma = 1e+06, priorVrho = "1/Gamma", shape = 0.5, rate = 0.0005, Vrho.max=1000, seed = 1234, verbose = 1, save.rho = 0, save.p = 0)

Arguments

counts: A vector indicating the count for each observation.
suitability: A one-sided formula of the form $\sim x_1 + \ldots + x_p$ with $p$ terms specifying the explicative variables for the suitability process.
abundance: A one-sided formula of the form $\sim w_1 + \ldots + w_q$ with $q$ terms specifying the explicative variables for the abundance process.
spatial.entity: A vector indicating the spatial entity identifier (from one to the total number of entities) for each observation. Several observations can occur in one spatial entity. A spatial entity can be a raster cell for example.
alteration: A vector indicating the proportion of area in the spatial cell which is transformed (by anthropogenic activities for example) for each observation. Must be between 0 and 1.
data: A data frame containing the model’s variables.
n.neighbors: A vector of integers that indicates the number of neighbors (adjacent entities) of each spatial entity. length(n.neighbors) indicates the total number of spatial entities.
neighbors: A vector of integers indicating the neighbors (adjacent entities) of each spatial entity. Must be of the form c(neighbors of entity 1, neighbors of entity 2, …, neighbors of the last entity). Length of the neighbors vector should be equal to sum(n.neighbors).
suitability.pred
   An optional data frame in which to look for variables with which to predict. If
   NULL, the observations are used.

spatial.entity.pred
   An optional vector indicating the spatial entity identifier (from one to the total
   number of entities) for predictions. If NULL, the vector spatial.entity for
   observations is used.

burnin
   The number of burnin iterations for the sampler.

mcmc
   The number of Gibbs iterations for the sampler. Total number of Gibbs iterations
   is equal to burnin+mcmc. burnin+mcmc must be divisible by 10 and superior or
   equal to 100 so that the progress bar can be displayed.

thin
   The thinning interval used in the simulation. The number of mcmc iterations
   must be divisible by this value.

beta.start
   Starting values for $\beta$ parameters of the suitability process. This can either be a
   scalar or a $p$-length vector.

gamma.start
   Starting values for $\beta$ parameters of the observability process. This can either be
   a scalar or a $q$-length vector.

Vrho.start
   Positive scalar indicating the starting value for the variance of the spatial random
   effects.

mubeta
   Means of the priors for the $\beta$ parameters of the suitability process. mubeta must
   be either a scalar or a $p$-length vector. If mubeta takes a scalar value, then that
   value will serve as the prior mean for all of the betas. The default value is set to
   0 for an uninformative prior.

Vbeta
   Variances of the Normal priors for the $\beta$ parameters of the suitability process.
   Vbeta must be either a scalar or a $p$-length vector. If Vbeta takes a scalar value,
   then that value will serve as the prior variance for all of the betas. The default
   variance is large and set to 1.0E6 for an uninformative flat prior.

mugamma
   Means of the Normal priors for the $\gamma$ parameters of the observability process.
   mugamma must be either a scalar or a $p$-length vector. If mugamma takes a scalar
   value, then that value will serve as the prior mean for all of the gammas. The
   default value is set to 0 for an uninformative prior.

Vgamma
   Variances of the Normal priors for the $\gamma$ parameters of the observability process.
   Vgamma must be either a scalar or a $p$-length vector. If Vgamma takes a scalar
   value, then that value will serve as the prior variance for all of the gammas. The
   default variance is large and set to 1.0E6 for an uninformative flat prior.

priorVrho
   Type of prior for the variance of the spatial random effects. Can be set to a fixed
   positive scalar, or to an inverse-gamma distribution ("1/Gamma") with param-
   eters shape and rate, or to a uniform distribution ("Uniform") on the interval
   [0,Vrho.max]. Default set to "1/Gamma".

shape
   The shape parameter for the Gamma prior on the precision of the spatial random
   effects. Default value is shape=0.05 for uninformative prior.

rate
   The rate (1/scale) parameter for the Gamma prior on the precision of the spatial
   random effects. Default value is rate=0.0005 for uninformative prior.

Vrho.max
   Upper bound for the uniform prior of the spatial random effect variance. Default
   set to 1000.
seed The seed for the random number generator. Default set to 1234.

verbose A switch (0,1) which determines whether or not the progress of the sampler is printed to the screen. Default is 1: a progress bar is printed, indicating the step (in %) reached by the Gibbs sampler.

save.rho A switch (0,1) which determines whether or not the sampled values for rhos are saved. Default is 0: the posterior mean is computed and returned in the rho.pred vector. Be careful, setting save.rho to 1 might require a large amount of memory.

save.p A switch (0,1) which determines whether or not the sampled values for predictions are saved. Default is 0: the posterior mean is computed and returned in the prob.p.pred vector. Be careful, setting save.p to 1 might require a large amount of memory.

Details

The model integrates two processes, an ecological process associated to the presence or absence of the species due to habitat suitability and an observation process that takes into account the fact that the probability of detection of the species is inferior to one. The ecological process includes an intrinsic conditional autoregressive model (iCAR) model for spatial autocorrelation between observations, assuming that the probability of presence of the species at one site depends on the probability of presence of the species on neighboring sites.

Ecological process:

\[ z_i \sim \text{Bernoulli}(\theta_i) \]

\[ \logit(\theta_i) = X_i \beta + \rho_{j(i)} \]

\( \rho_j \): spatial random effect

\( j(i) \): index of the spatial entity for observation \( i \).

Spatial autocorrelation:

An intrinsic conditional autoregressive model (iCAR) is assumed:

\[ \rho_j \sim \text{Normal}(\mu_j, V_\rho/n_j) \]

\( \mu_j \): mean of \( \rho_j \) in the neighborhood of \( j \).

\( V_\rho \): variance of the spatial random effects.

\( n_j \): number of neighbors for spatial entity \( j \).

Observation process:

\[ y_i \sim \text{Binomial}(z_i \ast \delta_i, t_i) \]

\[ \logit(\delta_i) = W_i \gamma \]

Value

mcmc An mcmc object that contains the posterior sample. This object can be summarized by functions provided by the coda package. The posterior sample of the deviance \( D \), with \( D = -2 \log(\prod_i P(y_i, z_i | \ldots)) \), is also provided.
rho.pred  If save.rho is set to 0 (default), rho.pred is the predictive posterior mean of
the spatial random effect associated to each spatial entity. If save.rho is set
to 1, rho.pred is an mcmc object with sampled values for each spatial random
effect associated to each spatial entity.

prob.p.pred  If save.p is set to 0 (default), prob.p.pred is the predictive posterior mean
of the probability associated to the suitability process for each prediction. If
save.p is set to 1, prob.p.pred is an mcmc object with sampled values of the
probability associated to the suitability process for each prediction.

prob.p.latent  Predictive posterior mean of the probability associated to the suitability process
for each observation.

prob.q.latent  Predictive posterior mean of the probability associated to the observability pro-
cess for each observation.

Author(s)
Ghislain Vieilledent <ghislain.vieilledent@cirad.fr>

References
Flores, O.; Rossi, V. and Mortier, F. (2009) Autocorrelation offsets zero-inflation in models of
tropical saplings density. Ecological Modelling, 220, 1797-1809.
Modelling species diversity through species level hierarchical modelling. Applied Statistics, 54,
1-20.

See Also
plot.mcmc, summary.mcmc

Examples

## Not run:

#===============================================
# hSDM.ZIP.iCAR.alteration()
# Example with simulated data
#===============================================

#-------
# Preambule
library(hSDM)
library(raster)
library(sp)
library(mvtnorm)

#-------
```r
### Data simulation

# Set seed for repeatability
seed <- 1234

# Target parameters
beta.target <- matrix(c(0.2, 0.5, 0.5), ncol=1)
gamma.target <- matrix(c(1), ncol=1)
## Uncomment if you want covariates on the observability process
## gamma.target <- matrix(c(0.2, 0.5, 0.5), ncol=1)
Vrho.target <- 1 # Spatial Variance

# Landscape
Landscape <- raster(ncol=20, nrow=20, crs='+proj=utm +zone=1')
ncell <- ncell(Landscape)

# Neighbors
neighbors.mat <- adjacent(Landscape, cells=c(1:ncell), directions=8, pairs=TRUE, sorted=TRUE)
n.neighbors <- as.data.frame(table(as.factor(neighbors.mat[,1])))[,2]
adj <- neighbors.mat[,2]

# Generate symmetric adjacency matrix, A
A <- matrix(0, ncell, ncell)
index.start <- 1
for (i in 1:ncell) {
  index.end <- index.start+n.neighbors[i]-1
  A[i, adj[c(index.start:index.end)]] <- 1
  index.start <- index.end+1
}

# Spatial effects
d <- 1 # Spatial dependence parameter = 1 for intrinsic CAR
Q <- diag(n.neighbors)-d*A + diag(.0001, ncell) # Add small constant to make Q non-singular
covrho <- Vrho.target*solve(Q) # Covariance of rhos
set.seed(seed)
rho <- crmvnorm(1, sigma=covrho) # Spatial Random Effects
rho <- rho-mean(rho) # Centering rhos on zero

# Visited cells
n.visited <- 150 # Compare with 400, 350 and 100 for example
set.seed(seed)
visited.cells <- sort(sample(1:ncell, n.visited, replace=FALSE)) # Draw visited cells at random
notvisited.cells <- c(1:ncell)[-visited.cells]

# Number of observations
nobs <- 300

# Cell vector
set.seed(seed)
cells <- c(visited.cells, sample(visited.cells, nobs-n.visited, replace=TRUE))
coords <- xyFromCell(Landscape, cells) # Get coordinates

# Covariates for "suitability" process
```
```r
set.seed(seed)
X1.cell <- rnorm(n=nCELL,0,1)
set.seed(2*seed)
X2.cell <- rnorm(n=nCELL,0,1)
X1 <- X1.cell[cells]
X2 <- X2.cell[cells]
X <- cbind(rep(1,nObs),X1,X2)

# Alteration
U <- runif(n=nObs,min=0,max=1)

# Covariates for "abundance" process
W <- cbind(rep(1,nObs))
## Uncomment if you want covariates on the observability process
## set.seed(3*seed)
## W1 <- rnorm(n=nObs,0,1)
## set.seed(4*seed)
## W2 <- rnorm(n=nObs,0,1)
## W <- cbind(rep(1,nObs),W1,W2)

##= Simulating latent variables

## Suitability
logit(theta) <- vector()
for (n in 1:nObs) {
  logit.theta[n] <- X[n,]*%*%beta+rho[cells[n]]
}
theta <- inv.logit(logit.theta)
set.seed(seed)
y.1 <- rbinom(n=nObs,1,theta)

# Alteration
u <- rbinom(n=nObs,1,U)

## Abundance
set.seed(seed)
log.lambda <- W %*% gamma.target
lambda <- exp(log.lambda)
set.seed(seed)
y.2 <- rpois(n=nObs,lambda)

##= Simulating response variable
Y <- y.2*(1-u)*y.1

## Data-set
Data <- data.frame(Y,cells,X1,X2,U)
## Uncomment if you want covariates on the observability process
## Data <- data.frame(Y,cells,X1,X2,W1,W2,U)
Data <- SpatialPointsDataFrame(coords=coords,data=Data)
plot(Data)

##= Data-set for predictions (suitability on each spatial cell)
Data.pred <- data.frame(X1=X1.cell,X2=X2.cell,cells=c(1:ncell))
```
# Site-occupancy model

```r
mod.hSDM.ZIP.iCAR.alteration <- hSDM.ZIP.iCAR.alteration(counts=Data$Y,
suitability=-X1+X2,
abundance=1,
spatial.entity=Data$cells,
alteration=Data$U,
data=Data,
n.neighbors=n.neighbors,
neighbors=adj,
## suitability.pred=NULL,
## spatial.entity.pred=NULL,
suitability.pred=Data.pred,
spatial.entity.pred=Data.pred$cells,
burnin=5000, mcmc=5000, thin=5,
beta.start=0,
gamma.start=0,
vrho.start=10,
priorVrho="1/Gamma",
#priorVrho="Uniform",
#priorVrho=10,
mubeta=0, Vbeta=1.0E6,
mugamma=0, Vgamma=1.0E6,
shape=0.5, rate=0.0005,
#Vrho.max=1000,
seed=1234, verbose=1,
save.rho=1, save.p=0)
```

# Outputs

## Parameter estimates

```r
summary(mod.hSDM.ZIP.iCAR.alteration$mcmc)
```

## MCMC and posteriors

```r
df <- "Posterior_hSDM.ZIP.iCAR.alteration.pdf"
pdf(mod.hSDM.ZIP.iCAR.alteration$mcmc)
dev.off()
```

```r
df <- "Posterior_rho_hSDM.ZIP.iCAR.alteration.pdf"
pdf(mod.hSDM.ZIP.iCAR.alteration$rho.pred)
dev.off()
```

## Summary plots

```r
r.rho <- r.rho.pred <- r.visited <- Landscape
r.rho[] <- rho
rho.pred <- apply(mod.hSDM.ZIP.iCAR.alteration$rho.pred,2,mean)
r.rho.pred[] <- rho.pred
r.visited[] <- 0
```
r.visited[visited.cells] <- tapply(Data$Y,Data$cells,mean)
# prob.p
r.prob.p <- Landscape
r.prob.p[] <- mod.hSDM.ZIP.iCAR.alteration$prob.p.pred

pdf(file="Summary_hSDM.ZIP.iCAR.alteration.pdf")
par(mfrow=c(3,2))
plot(r.rho, main="rho target")
plot(r.visited,main="Visited cells and counts")
plot(Data,add=TRUE,pch=16,cex=0.5)
plot(r.rho.pred, main="rho estimated")
plot(rho[visited.cells],rho.pred[visited.cells],
     xlab="rho target",
ylab="rho estimated")
points(rho[notvisited.cells],rho.pred[notvisited.cells],pch=16,col="blue")
legend(x=-4,y=3.5,legend="Unvisited cells",col="blue",pch=16,bty="n")
abline(a=0,b=1,col="red")
plot(r.prob.p,main="Predicted counts")
plot(Data,add=TRUE,pch=16,cex=0.5)
dev.off()

## End(Not run)

---

### logit

**Generalized logit and inverse logit function**

#### Description

Compute generalized logit and generalized inverse logit functions.

#### Usage

```r
logit(x, min = 0, max = 1)
inv.logit(x, min = 0, max = 1)
```

#### Arguments

- `x`: value(s) to be transformed
- `min`: Lower end of logit interval
- `max`: Upper end of logit interval

#### Details

The generalized logit function takes values on [min, max] and transforms them to span [-Inf,Inf]. It is defined as:

\[ y = \log \left( \frac{p}{1 - p} \right) \]
where

\[ p = \frac{(x - \min)}{(\max - \min)} \]

The generized inverse logit function provides the inverse transformation:

\[ x = p'(\max - \min) + \min \]

where

\[ p' = \frac{\exp(y)}{(1 + \exp(y))} \]

Value

Transformed value(s).

Author(s)

Gregory R. Warnes <greg@warnes.net>

Examples

```r
## Not run:
x <- seq(0, 10, by=0.25)
x <- logit(x, min=0, max=10)
cbind(x,x)

y <- inv.logit(x, min=0, max=10)
cbind(x,x,y)

## End(Not run)
```

Description

Data come from a small region including 476 one minute by one minute grid cells. This region is a small corner of South Africa’s Cape Floristic Region, and includes very high plant species diversity and a World Biosphere Reserve. The data frame can be used as an example for several functions in the hSDM package.

Usage

neighbors.Latimer2006
Format

neighbors.Latimer2006 is a vector of 3542 integers indicating the neighbors (adjacent cells) of each spatial cell. The vector is of the form c(neighbors of cell 1, neighbors of cell 2, ..., neighbors of the last cell).

Source

Latimer et al. (2006) *Ecological Applications*, Appendix B

References


**punc10**

Occurrence data for *Protea punctata* Meisn. in the Cap Floristic Region

**Description**

The species data were collected by the Protea Atlas Project of South Africa’s National Botanical Institute.

**Usage**

punc10

**Format**

cfr.env is a data frame with 2934 presence-absence observation points.

0ccurrence presence (1) or absence (0) of the species
lon longitude
lat latitude

**Source**

Cory Merow’s personal data

References

Index

*Topic Binomial logistic regression
  hSDM.binomial, 8
  hSDM.binomial.iCAR, 11

*Topic MCMC
  hSDM.package, 2
  hSDM.binomial, 8
  hSDM.binomial.iCAR, 11
  hSDM.Nmixture, 18
  hSDM.Nmixture.iCAR, 23
  hSDM.Nmixture.K, 32
  hSDM.poisson, 37
  hSDM.poisson.iCAR, 40
  hSDM.siteocc, 47
  hSDM.siteocc.iCAR, 51
  hSDM.ZIB, 60
  hSDM.ZIB.iCAR, 64
  hSDM.ZIB.iCAR.alteration, 71
  hSDM.ZIP, 79
  hSDM.ZIP.iCAR, 83
  hSDM.ZIP.iCAR.alteration, 91

*Topic Markov Chains Monte Carlo
  hSDM.package, 2
  hSDM.binomial, 8
  hSDM.binomial.iCAR, 11
  hSDM.Nmixture, 18
  hSDM.Nmixture.iCAR, 23
  hSDM.Nmixture.K, 32
  hSDM.poisson, 37
  hSDM.poisson.iCAR, 40
  hSDM.siteocc, 47
  hSDM.siteocc.iCAR, 51
  hSDM.ZIB, 60
  hSDM.ZIB.iCAR, 64
  hSDM.ZIB.iCAR.alteration, 71
  hSDM.ZIP, 79
  hSDM.ZIP.iCAR, 83
  hSDM.ZIP.iCAR.alteration, 91

*Topic Metropolis algorithm
  hSDM.package, 2

*Topic N-mixture models
  hSDM.Nmixture, 18
  hSDM.Nmixture.iCAR, 23
  hSDM.Nmixture.K, 32

*Topic Poisson log regression
  hSDM.poisson, 37

*Topic Poisson regression
  hSDM.poisson, 37

*Topic ZIB models
  hSDM.ZIB, 60
  hSDM.ZIB.iCAR, 64
  hSDM.ZIB.iCAR.alteration, 71
  hSDM.ZIP, 79
  hSDM.ZIP.iCAR, 83
  hSDM.ZIP.iCAR.alteration, 91

*Topic ZIP models
  hSDM.ZIP, 79
  hSDM.ZIP.iCAR, 83

*Topic Zero-Inflated Binomial models
  hSDM.ZIB, 60
  hSDM.ZIB.iCAR, 64
  hSDM.ZIB.iCAR.alteration, 71
  hSDM.ZIP, 79
  hSDM.ZIP.iCAR, 83
  hSDM.ZIP.iCAR.alteration, 91

*Topic Zero-Inflated Poisson models
  hSDM.ZIP, 79
  hSDM.ZIP.iCAR, 83
  hSDM.ZIP.iCAR.alteration, 91

*Topic biodiversity
hsDM-package, 2
hsDM.binomial, 8
hsDM.binomial.icar, 11
hsDM.Nmixture, 18
hsDM.Nmixture.icar, 23
hsDM.Nmixture.K, 32
hsDM.poisson, 37
hsDM.poisson.icar, 40
hsDM.siteocc, 47
hsDM.siteocc.icar, 51
hsDM.ZIB, 60
hsDM.ZIB.icar, 64
hsDM.ZIB.icar.alteration, 71
hsDM.ZIP, 79
hsDM.ZIP.icar, 83
hsDM.ZIP.icar.alteration, 91

*Topic **conditional autoregressive model**
hsDM-package, 2
hsDM.binomial.icar, 11
hsDM.Nmixture.icar, 23
hsDM.poisson.icar, 40
hsDM.siteocc.icar, 51
hsDM.ZIB.icar, 64
hsDM.ZIB.icar.alteration, 71
hsDM.ZIP.icar, 83
hsDM.ZIP.icar.alteration, 91

*Topic **datasets**
altitude, 3
cfr.env, 4
data.Kery2010, 5
datacells.Latimer2006, 6
frogs, 7
neighbors.Latimer2006, 99
punct10, 100

*Topic **hierarchical Bayesian models**
hsDM-package, 2
hsDM.binomial, 8
hsDM.binomial.icar, 11
hsDM.Nmixture, 18
hsDM.Nmixture.icar, 23
hsDM.Nmixture.K, 32
hsDM.poisson, 37
hsDM.poisson.icar, 40
hsDM.siteocc, 47
hsDM.siteocc.icar, 51
hsDM.ZIB, 60
hsDM.ZIB.icar, 64
hsDM.ZIB.icar.alteration, 71
hsDM.ZIP, 79
hsDM.ZIP.icar, 83
hsDM.ZIP.icar.alteration, 91

*Topic **intrinsic CAR model**
hsDM-package, 2
hsDM.binomial.icar, 11
hsDM.Nmixture.icar, 23
hsDM.poisson.icar, 40
hsDM.siteocc.icar, 51
hsDM.ZIB.icar, 64
hsDM.ZIB.icar.alteration, 71
hsDM.ZIP.icar, 83
hsDM.ZIP.icar.alteration, 91

*Topic **math**
logit, 98

*Topic **site occupancy models**
hsDM.siteocc.icar, 51
hsDM.ZIP.icar, 83
hsDM.ZIP.icar.alteration, 91

*Topic **site-occupancy models**
hsDM.siteocc, 47

*Topic **spatial correlation**
hsDM-package, 2
hsDM.binomial.icar, 11
hsDM.Nmixture.icar, 23
hsDM.poisson.icar, 40
hsDM.siteocc.icar, 51
hsDM.ZIB.icar, 64
hsDM.ZIB.icar.alteration, 71
hsDM.ZIP.icar, 83
hsDM.ZIP.icar.alteration, 91

*Topic **species distribution models**
hsDM-package, 2
hsDM.binomial, 8
hsDM.binomial.icar, 11
hsDM.Nmixture, 18
hsDM.Nmixture.icar, 23
hsDM.Nmixture.K, 32
hsDM.poisson, 37
hsDM.poisson.icar, 40
hsDM.siteocc, 47
hsDM.siteocc.icar, 51
hsDM.ZIB, 60
hsDM.ZIB.icar, 64
hsDM.ZIB.icar.alteration, 71
hsDM.ZIP, 79
hsDM.ZIP.icar, 83
INDEX

hSDM.ZIP.iCAR.alteration, 91
altitude, 3
cfr.env, 4
data.Kery2010, 5
datacells.Latimer2006, 6
frogs, 7
hSDM (hSDM-package), 2
hSDM-package, 2
hSDM.binomial, 8
hSDM.binomial.iCAR, 11
hSDM.Nmixture, 18
hSDM.Nmixture.iCAR, 23
hSDM.Nmixture.K, 32
hSDM.poisson, 37
hSDM.poisson.iCAR, 40
hSDM.siteocc, 47
hSDM.siteocc.iCAR, 51
hSDM.ZIB, 60
hSDM.ZIB.iCAR, 64
hSDM.ZIB.iCAR.alteration, 71
hSDM.ZIP, 79
hSDM.ZIP.iCAR, 83
hSDM.ZIP.iCAR.alteration, 90
inv.logit (logit), 98
logit, 98
neighbors.Latimer2006, 99
plot.mcmc, 10, 14, 21, 27, 35, 39, 43, 49, 56, 62, 68, 75, 81, 87, 94
punct0, 100
summary.mcmc, 10, 14, 21, 27, 35, 39, 43, 49, 56, 62, 68, 75, 81, 87, 94