Package 'stocc'

October 14, 2022

Type Package

Title Fit a Spatial Occupancy Model via Gibbs Sampling

Version 1.31

Date 2021-05-10

Author Devin S. Johnson

Maintainer Devin S. Johnson <devin.johnson@noaa.gov>

Description Fit a spatial-temporal occupancy models using a probit formulation instead of a traditional logit model.

License CC0

Imports truncnorm, coda, Matrix, fields, rARPACK

LazyLoad yes

RoxygenNote 7.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2021-05-11 17:30:06 UTC

R topics documented:

`stocc-package` *Fit a Spatial Occupancy Model via Gibbs Sampling*

Description

This package contains functions that fit a spatial occupancy model where the true occupancy is a function of a spatial process. An efficient Gibbs sampling algorithm is used by formulating the detection and occupancy process models with a probit model instead of the traditional logit based model.

Details

Package:	stocc
Type:	Package
Version:	1.4
Date:	May 10, 2021
License:	CC0
LazyLoad:	yes

Author(s)

Devin S. Johnson

Maintainer: Devin S. Johnson <devin.johnson@noaa.gov>

`habData` *A simulated data set of environmental covariates*

Description

This data represents a simulated study area. The study area is a 40 x 40 grid of pixels. There are two variables, a factor variable (e.g., a habitat layer), as well as, a continuous covariate.

Format

A data frame with 1600 observations on the following 5 variables.

site Site labels

x Longitude coordinate

y Latitude coordinate

habCov1 a factor with levels `1` `2` `3`

habCov2 a numeric vector

Examples

```
data(habData)
image(x=seq(0.5,39.5,1), y=seq(0.5,39.5,1),
z=t(matrix(as.numeric(habData$habCov1),40)), main="habData: Factor environmental covariate",
xlab="x", ylab="y", col=rainbow(3))


dev.new()
image(x=seq(0.5,39.5,1), y=seq(0.5,39.5,1),
z=t(matrix(habData$habCov2,40)), main="habData: Continuous environmental covariate",
xlab="x", ylab="y", col=terrain.colors(50))
```

`icar.Q` *Creates the inverse covariance matrix for an intrinsic conditionally autoregressive spatial model.*

Description

This function creates the ICAR precision matrices used in the spatial models

Usage

```
icar.Q(xy, threshold, rho = 1, fun = FALSE)
```

Arguments

`xy`	An n x 2 matrix of spatial coordinates
`threshold`	Distance threshold for neighborhood definition
`rho`	The autoregressive parameter. Defaults to 1, which is the Intrinsic Conditionally AutoRegressive model (ICAR)
`fun`	If `TRUE` this function returns a function of rho that generates the precision matrix of a ICAR process

Details

Constructs the inverse covariance matrix (aside from scaling) for the ICAR model

Value

An n x n matrix

Author(s)

Devin S. Johnson <devin.johnson@noaa.gov>

`make.so.data` *Create spatial data object for model fitting via* `spatialOccupancy` *function*

Description

This function takes an observation data frame and a data frame of site characteristics and combines them together for analysis with the `spatial.occupancy` function.

Usage

```
make.so.data(visit.data, site.data, names)
```

Arguments

`visit.data`	A data frame that contains the observed occupancy for each site and any detection related covariates.
`site.data`	A data frame that contains the site id, coordinates, and any habitat related covariates that might influence the occupancy process
`names`	A named list with the following elements: (1)`visit` A named list with elements "site" = the name of the site id in the observation data frame and "obs" = the name of the observed occupancy variable (2) `site` A named list with elements "site" = the name of the site id and "coords" = a character vector giving the name of the coordinates (x first then y)

Details

This function combines the two data frames and assigns names so that `spatial.occupancy` knows which columns to use. It also performs some rudimentary error checking to make sure the data is in the proper form (e.g., the site IDs in the visit data frame must be contained in the site IDs for the site data frame)

Value

An `so.data` object is a list with elements equal to the two data frames. Attributes are set giving the names of columns of interest

Author(s)

Devin S. Johnson <devin.johnson@noaa.gov>

`occupancyData` *Simulated occupancy for the 40 x 40 study area.*

Description

This data represnts truth with regards to occupancy in the simulated study area. The probability of occupancy was simulated as `pnorm(0, X + K alpha, 1, lower=FALSE)`, where `K` and `alpha` were constructed from a reduced rank is an ICAR process with precision (`tau`) = 0.3 and `gamma = c(-1, 0, 0, 1)`

Format

A data frame with 1600 observations on the following 5 variables.

site Site labels

x Longitude coordinate

y Latitude coordinate

psi True probability of occupancy

psi.fix The fixed effects portion of the occupancy process map

occ True realized occupancy

Examples

```
data(occupancyData)
##
## Blue points represent realized occupancy.
##
image(x=seq(0.5,39.5,1), y=seq(0.5,39.5,1), z=t(matrix(occupancyData$psi,40)),
xlab="x", ylab="y", main="Occupancy process with realized occupancy")
points(occupancyData$x[occupancyData$occ==1], occupancyData$y[occupancyData$occ==1],
 pch=20, cex=0.25, col="blue")
```

`spatial.occupancy` *Fit a spatial occupancy model using Gibbs sampling*

Description

This function fits a spatial occupancy model where the true occupancy is a function of a spatial process. An efficient Gibbs sampling algorithm is used by formulating the detection and occupancy process models with a probit model instead of the traditional logit based model.

Usage

```
spatial.occupancy(
  detection.model,
  occupancy.model,
  spatial.model,
  so.data,
  prior,
  control,
  initial.values = NULL
)
```

Arguments

`detection.model`
: A formula object describing the detection portion of the occupancy model. The variables described by the detection model are located in the `visit` data frame of the `so.data`.

`occupancy.model`
: A formula object describing the fixed effects portion of the spatial occupancy process. The variables described by the occupancy model are located in the `site` data frame of an `so.data` object.

`spatial.model`
: A named list object describing the spatial component of the occupancy process. Currently the only possible models are ICAR, restricted spatial regression, process convolution models, and no spatial model (i.e., eta = 0). Thus, `spatial.model=list(model="icar",threshold= )`, `spatial.model=list(model="rsr", threshold=, moran.cut=)`, `spatial.model=list(model="proc.conv", knots=)`, and `spatial.model=list(model="none")` are the only forms that are accepted at present. The `threshold` component is used the create neighborhoods in the ICAR and RSR models. All sites within distance `threshold` of site i are considered neighbors of site i. The `moran.cut` component is the cut-off for selecting the spatial harmonics used in the restricted spatial regression model. The value must be between 1 and N and implies that the eigen vectors associated with the largest `moan.cut` eigen values are used for the basis functions. The item `knots` are xy locations of the discrete process convolution knots.

`so.data`
: An `so.data` object containing the observed occupancies, detection covariates, site covariates, and site coordinates. This is created via the `make.so.data`

`prior`
: A named list that provides the parameter values for the prior distributions. At the current time the elements of the list must contain `a.tau` and `b.tau` which are the parameters for the gamma prior on the spatial process parameter in the occupancy model. Other elements may include `Q.b` and `mu.b` which are the tolerance and mean for the beta vector (detection parameters). Also `Q.g` and `mu.g` which are the prior parameters for the occupancy model. If the Q.b and Q.g are left out, the default is Q.b = 0 and Q.g = 0 (i.e., flat priors). If mu.b and mu.g are left out, the default is zero vectors.

`control`
: A named list with the control parameters for the MCMC. The elements of the list must include: (1) `burnin` is the number of iterations of burnin, (2) `iter` is the total number of iterations retained for the MCMC sample, and (3) `thin` is

the thining rate of the chain. The real number of MCMC iterations is equal to `iter*thin` of which `iter - burnin` are retained for posterior summary.

`initial.values` A named list that can include any or all of the following vectors or scalers (1) `beta`, a vector of initial values for the detection parameters, (2) `gamma`, a vector or initial values for the occupancy model, and (3) `tau`, an initial value for the spatial precision parameter.

Details

A Gibbs sampler is run to draw an MCMC sample of the spatial occupancy parameters `beta` (detection parameters), `gamma` (the occupancy parameters), `psi` (the model occupancy generating process), and the realized occupancy.

Value

A list with the following elements:

`beta` An object of class `mcmc`. The detection model parameters.

`gamma` An object of class `mcmc`. The occupancy model parameters.

`psi` An object of class `mcmc`. The occupancy generating process

`real.occ` An object of class `mcmc`. The realized occupancy at the time of the survey

`tau` An object of class `mcmc`. The variance parameter for the spatial model

`occupancy.df` A data frame with the spatial coordinates, site id, and posterior mean and variance of psi, eta, and real.occ

`D.m` The posterior predictive loss criterion of Gelfand and Ghosh (1998; Biometrika 85:1-11) for model selection. The criterion is a combination of a goodness-of-fit measure, G.m, and a complexity measure, P.m, similar information criteria such as AIC and BIC. D.m = G.m + P.m. Lower values of D.m imply lower expected loss in predicting new data with the posterior model parameters.

`G.m` The goodness-of-fit portion of D.m

`P.m` The model complexity component of D.m

`detection.model` The detection model call.

`occupancy.model` The occupancy model call.

`model` A character version of the joint occupancy and detection model call. This is useful for saving results.

Author(s)

Devin S. Johnson <devin.johnson@noaa.gov>

visitData *Simulated occupancy survey data*

Description

Data set representing a simulated survey of the 40 x 40 study area. Approximately 1/3 of the 1600 sites were visited at least once. Those sites that were surveyed were visited a random number of times with an average of 2.5 visits. Detection was simulated as a function of 2 covariates, a continuous one (cov1) and a factor (cov2). These are NOT the same as the cov1 and cov2 of the habData data frame. The coefficients used were `beta = c(1, 0, 0.5, 1, 0)`. Thus detection given occupancy of site i at time j = `pnorm(0,X%*%beta,lower=FALSE)`.

Format

A data frame with 1340 observations on the following 6 variables.

site Site labels

x Longitude coordinate

y Latitude coordinate

detCov1 a numeric vector

detCov2 a factor with levels `0 1 2 3`

obs a numeric vector

Examples

```
data(visitData)
data(occupancyData)
##
## Blue points represent visited sites and green circles represent confirmed occupancy.
##
image(x=seq(0.5,39.5,1), y=seq(0.5,39.5,1), z=t(matrix(occupancyData$psi,40)),
xlab="x", ylab="y", main="Occupancy process with visits")
points(visitData$x[visitData$obs==1], visitData$y[visitData$obs==1], col="green")
points(visitData$x, visitData$y, col="blue", pch=20, cex=0.25)
```

Index