popdynmodel

An R package for modelling population dynamics and demography

November 28, 2023

Contents

1. Installation

popdynmodel depends on R $\geq$ 4.2.0 and requires installing and loading the following packages:

```
library(nimble)
library(MCMCvis)
library(rlang)
library(magrittr)
library(dplyr)
library(tidyr)
library(stringr)
library(tidyverse)
```

Then the file containing the package functions must be loaded:

```
source("Fonction_package_modelisation.R")
```

For the get_modparameters function, the following file must also be loaded:

```
load("modfunparameters.rda")
```

2. Package content

riverfish

Description

This dataset gives the abundance and biomass of seven freshwater fish species on 85 sites sampled, from 2010 to 2022, by multiple-pass removal electrofishing. The sites are spread over six river-basins of metropolitan France.

Usage

```
data(riverfish)
```

Format

`riverfish` is a data frame with 14 variables:

`pop_id`	sampling site identifiers
`year`	sampling year
`taxa`	species common names
`family`	species family names
`gudgeon`	gudgeon guild: value is 1 if the species is a gudgeon and NA otherwise
`pass`	pass position in multi-pass removal electrofishing
`headcount`	fishing catch size by species
`biomass`	fishing catch weight by species
`surface`	fishing surface
`fishing`	electrofishing protocol
`hydro_basin`	river-basin identifiers
`temperature`	water temperature per site and year
`depth`	fishing depth per site and year
`flow`	watercourse flow per site and year

Source

`riverfish` is based on data extracted from the national ASPE database managed by the French Biodiversity Agency (OFB).

get_modparameters

Description

Gives the description of the parameters of modeling functions.

Usage

```
get_modparameters(fun, parameter)
```

Arguments

fun   [optional, character] name of the modelling function

parameter   [optional, character] name(s) of the parameter(s)

Examples

```
# Get description about all parameters of mod_popoccup function
get_modparameters(fun = "mod_popoccup")

# Get description about some parameters
get_modparameters(parameter = "z")
get_modparameters(parameter = c("z","N","B"))
```

mod_popoccup

Description

Fits Bayesian hierarchical occupancy models to presence-absence time series for estimating the occurrence of one or more taxa from colonisation and extinction probabilities. The modelling framework not accounts for taxa detection biases.

Usage

```
mod_popoccup(
    df,
    var_id,
    var_tmp,
    var_cnt,
    var_tax = NULL,
    var_reg = NULL,
    var_guild = NULL,
    period = NULL,
    timestep = 1,
    save_parameters = NULL
    n_chain = 3,
    n_iter = 10000,
    n_thin = n_iter / 100,
    n_burnin = n_iter / 4
)
```

Arguments

`df`	data.frame with variables used in the model
`var_id`	[character] variable name with sampling location identifiers (must not contain missing values)
`var_tmp`	[character] variable name with continuous sampling times (must be numeric without missing values)
`var_cnt`	[character] variable name used to define presence/absence (must be numeric)
`var_tax`	[optional, character] variable name with taxa identifiers to be specified for the multi-taxa models (must not contain missing values)
`var_reg`	[optional, character] variable name(s) with spatial level identifiers. If supplied, estimates of occupancy changes rates at intermediate spatial levels are added to the model. Otherwise, occupancy change rates are only estimated across all locations

var_guild	[optional, character] variable name(s) with taxa guilds identifiers. If supplied, estimates of guild occupancy changes rates are added to the model
period	[optional, numeric list] list with start and end times for estimating intermediate average occupancy change rates
timestep	[optional, numeric] interval between two consecutive times to account for regular gaps in time series in the rate estimates
save_parameters	[optional, character] name(s) of the parameter(s) to save (by default, all model parametes are saved)
n_chain	[optional, numeric] number of MCMC chains
n_iter	[optional, numeric] number of iterations per MCMC chain
n_thin	[optional, numeric] backup interval for MCMC samples
n_burnin	[optional, numeric] number of initial iterations to discard per Markov chain

Details

mod_popoccup relies on Bayesian hierarchical state-space models that describe the occupancy dynamics of one or more taxa from presence-absence data. The successive occupancy states of locations defined by var_id depend on taxon- and location-specific probabilities of persistence and colonisation. At the global level (i.e. across all locations) and at intermediate spatial levels defined by var_reg, the taxa occupancy changes rates are estimated from the ratio of the number of occuped location from one time to the next. Their geometric mean is used as average occupancy rate over the whole time modelling period and over intermediate time periods defined by period. If var_guild is supplied, the guild occupancy change rates are assessed from the geometric means of taxa occupancy rates.

The models are run using the runMCMC function of nimble package.

Value

mod_popoccup returns a list with the summaries of main statistics associated with the estimation of the requested parameters (mcmc_summary), the MCMC chains (mcmc_chain) and the list of parameters subscripts (subscript).

See Also

mod_popoccupRS and get_modparametes

Examples

```
## Not run:
data(riverfish)
df <- riverfish[riverfish$pass == 1,]

## Occupancy dynamics of eel
mcmc.out <- mod_popoccup(df[df$taxa == "eel",],
                         var_id = "pop_id",
                         var_tmp = "year",
                         var_cnt = "headcount")

## Occupancy dynamics of taxa and families and basin level
mcmc.out <- mod_popoccup(df,
                         var_id = "pop_id",
                         var_tmp = "year",
                         var_tax = "taxa",
                         var_cnt = "headcount",
                         var_reg = "hydro_basin",
                         var_guild = "family")

## With intermediate rates every 6 years
mcmc.out <- mod_popoccup(df,
                         var_id = "pop_id",
                         var_tmp = "year",
                         var_tax = "taxa",
                         var_cnt = "headcount",
                         var_reg = "hydro_basin",
                         var_guild = "family",
                         period = list(c(2010,2016),c(2016,2022)))

## End(Not run)
```

modenv_popoccup

Description

Fits Bayesian hierarchical occupancy models to presence-absence time series for estimating the occurrence of one or more taxa from relationships with environmental covariates. The modelling framework not accounts for taxa detection biases.

Usage

```
modenv_popoccup(
    df,
    var_id,
    var_tmp,
    var_cnt,
    var_tax = NULL,
    var_env0 = NULL,
    var_envP = NULL,
    var_envC = NULL,
    var_reg = NULL,
    var_guild = NULL,
    period = NULL,
    timestep = 1,
    save_parameters = NULL,
    n_chain = 3,
    n_iter = 10000,
    n_thin = n_iter/100,
    n_burnin = n_iter/4
)
```

Arguments

df	data.frame with variables used in the model
var_id	[character] variable name with sampling location identifiers (must not contain missing values)
var_tmp	[character] variable name with continuous sampling times (must be numeric without missing values)
var_cnt	[character] variable name used to define presence/absence (must be numeric)
var_tax	[optional, character] variable name with taxa identifiers to be specified for the multi-taxa models (must not contain missing values)
var_env0	[optional, character] environmental covariate(s) name(s) related to occupancy probabilities whose values must be supplied by location and/or by time (must be numeric without missing values)

var_envP	[optional, character] environmental covariate(s) name(s) related to persistence probabilities whose values must be supplied by location and/or by time (must be numeric without missing values)
var_envC	[optional, character] environmental covariate(s) name(s) related to colonisation probabilities whose values must be supplied by location and/or by time (must be numeric without missing values)
var_reg	[optional, character] variable name(s) with spatial level identifiers. If supplied, estimates of occupancy changes rates at intermediate spatial levels are added to the model. Otherwise, occupancy change rates are only estimated across all locations
var_guild	[optional, character] variable name(s) with taxa guilds identifiers. If supplied, estimates of guild occupancy changes rates are added to the model
period	[optional, numeric list] list with start and end times for estimating intermediate average occupancy change rates
timestep	[optional, numeric] interval between two consecutive times to account for regular gaps in time series in the rate estimates
save_parameters	[optional, character] name(s) of the parameter(s) to save (by default, all model parametes are saved)
n_chain	[optional, numeric] number of MCMC chains
n_iter	[optional, numeric] number of iterations per MCMC chain
n_thin	[optional, numeric] backup interval for MCMC samples
n_burnin	[optional, numeric] number of initial iterations to discard per Markov chain

Details

modenv_popoccup relies on Bayesian hierarchical state-space models that describe the occupancy dynamics of one or more taxa from presence-absence data. The successive occupancy states of locations defined by var_id depend on taxon-specific responses to environmental covariates. At the global level (i.e. across all locations) and at intermediate spatial levels defined by var_reg, the taxa occupancy changes rates are estimated from the ratio of the number of occuped location from one time to the next. Their geometric mean is used as average occupancy rate over the whole time modelling period and over intermediate time periods defined by period. If var_guild is supplied, the guild occupancy change rates are assessed from the geometric means of taxa occupancy rates.

If `var_env0` is supplied, the occupancy states of the locations are modelled from the linear effect of environmental covariates on occupancy probabilities.

If `var_envP` and `var_envC` are supplied, the occupancy states of the locations are modelled from the linear effect of environmental covariates on persistence and colonisation probabilities.

If only `var_envP` is supplied, the occupancy states of the locations are modelled from the linear effect of environmental covariates on persistence probabilities and from taxon- and location-specific colonisation probabilities.

If only `var_envC` is supplied, the occupancy states of the locations are modelled from the linear effect of environmental covariates on colonisation probabilities and from taxon- and location-specific persistence probabilities.

The models are run using the `runMCMC` function of `nimble` package.

Value

`modenv_popoccup` returns a list with the summaries of main statistics associated with the estimation of the requested parameters (`mcmc_summary`), the MCMC chains (`mcmc_chain`) and the list of parameters subscripts (`subscript`).

See Also

`modenv_popoccupRS` and `get_modparametes`

Examples

```
## Not run:
data(riverfish)
df <- riverfish[riverfish$pass == 1,]

# Occupancy dynamics of eel with linear environmental effects on occupancy probability
mcmc.out <- modenv_popoccup(df[df$taxa == "eel",],
                            var_id = "pop_id",
                            var_tmp = "year",
                            var_cnt = "headcount",
                            var_env0 = c("temperature","depth"))

# With intermediate rates every 6 years at basin level
mcmc.out <- modenv_popoccup(df[df$taxa == "eel",],
                            var_id = "pop_id",
                            var_tmp = "year",
                            var_cnt = "headcount",
                            var_env0 = c("temperature","depth"),
                            var_reg = "hydro_basin",
                            period = list(c(2010,2016),c(2016,2022)))
```

```
# Multi-taxa occupancy dynamics with linear environmental effects on persistence probabi
mcmc.out <- modenv_popoccup(df,
                              var_id = "pop_id",
                              var_tmp = "year",
                              var_cnt = "headcount",
                              var_tax = "taxa",
                              var_envP = c("temperature","depth"))

# Multi-taxa occupancy dynamics with linear environmental effects on persistence
# and colonisation probabilities
mcmc.out <- modenv_popoccup(df,
                              var_id = "pop_id",
                              var_tmp = "year",
                              var_cnt = "headcount",
                              var_tax = "taxa",
                              var_envP = c("temperature","depth"),
                              var_envC = "temperature")

# Add occupancy change rates of families
mcmc.out <- modenv_popoccup(df,
                              var_id = "pop_id",
                              var_tmp = "year",
                              var_cnt = "headcount",
                              var_tax = "taxa",
                              var_envP = c("temperature","depth"),
                              var_envC = "temperature",
                              var_guild = "family")

## End(Not run)
```

mod_popgrow

Description

Fits Bayesian hierarchical single or multi-species population growth models to abundance and/or biomass time series. The modelling framework not accounts for detection biases of taxa and individuals.

Usage

```
mod_popgrow(
    df,
    var_id,
    var_tmp,
    var_tax = NULL,
    var_cnt = NULL,
    var_wei = NULL,
    var_surf = NULL,
    var_reg = NULL,
    var_guild = NULL,
    period = NULL,
    timestep = 1,
    save_parameters = NULL,
    n_chain = 3,
    n_iter = 10000,
    n_thin = n_iter/100,
    n_burnin = n_iter/4
)
```

Arguments

df	data.frame with variables used in the model
var_id	[character] variable name with sampling location identifiers (must not contain missing values)
var_tmp	[character] variable name with continuous sampling times (must be numeric without missing values)
var_tax	[optional, character] variable name with taxa identifiers to be specified for the multi-taxa models (must not contain missing values)
var_cnt	[optional, character] variable name with taxa abundances (must be numeric)
var_wei	[optional, character] variable name with taxa biomasses (must be numeric)
var_surf	[optional, character] variable name with quantitative sampling effort (e.g. sampling surfaces). If supplied, the model takes into account

variabilities in sampling effort from one time to the next
(must be numeric without missing values)

var_reg   [optional, character] variable name(s) with spatial level identifiers.
                   If supplied, estimates of occupancy changes rates at intermediate
                   spatial levels are added to the model. Otherwise, occupancy change
                   rates are only estimated across all locations

var_guild   [optional, character] variable name(s) with taxa guilds identifiers. If
                   supplied, estimates of guild occupancy changes rates are added to
                   the model

period   [optional, numeric list] list with start and end times for estimating
                   intermediate average growth rates

timestep   [optional, numeric] interval between two consecutive times to account
                   for regular gaps in time series in the rate estimates

save_parameters   [optional, character] name(s) of the parameter(s) to save (by default,
                   all model parametes are saved)

n_chain   [optional, numeric] number of MCMC chains

n_iter   [optional, numeric] number of iterations per MCMC chain

n_thin   [optional, numeric] backup interval for MCMC samples

n_burnin   [optional, numeric] number of initial iterations to discard per Markov
                   chain

Details

mod_popgrow relies on Bayesian hierarchical state-space models that include two or three main components. The first component describes the occupancy dynamics of taxa at locations defined by var_id from taxon- and location-specific probabilities of persistence and colonisation. Two other independent components describe the population growth of taxa on successively occupied locations, one based on abundance (if var_cnt is supplied) and other based on biomass (if var_wei is supplied). These two components are based on an autoregressive growth model. At location, global (i.e. across all locations) and intermediate spatial levels defined by var_reg, the finite population growth rates are estimated as the changes in population size from one time to the next. Their geometric means are used as average growth rates over the whole time modelling period and over intermediate time periods defined by period. If var_guild is supplied, the guild occupancy growth rates are assessed from the geometric means of taxa growth rates.

The models are run using the runMCMC function of nimble package.

Value

`mod_popgrow` returns a list with the summaries of main statistics associated with the estimation of the requested parameters (`mcmc_summary`), the MCMC chains (`mcmc_chain`) and the list of parameters subscripts (`subscript`).

See Also

`mod_popgrowRS` and `get_modparametes`

Examples

```
## Not run:
data(riverfish)
df <- riverfish[riverfish$pass == 1,]


# Pike population growth from abundance
mcmc.out <- mod_popgrow(df[df$taxa == "pike",],
                        var_id = "pop_id",
                        var_tmp = "year",
                        var_cnt = "headcount")


# Multi-taxa population growth from biomass
mcmc.out <- mod_popgrow(df,
                        var_id = "pop_id",
                        var_tmp = "year",
                        var_tax = "taxa",
                        var_wei = "biomass")


# Multi-taxa population growth from abundance and biomass
mcmc.out <- mod_popgrow(df,
                        var_id = "pop_id",
                        var_tmp = "year",
                        var_tax = "taxa",
                        var_cnt = "headcount",
                        var_wei = "biomass")


# Account for variabilities in annual sampling surfaces
mcmc.out <- mod_popgrow(df,
                        var_id = "pop_id",
                        var_tmp = "year",
                        var_tax = "taxa",
                        var_cnt = "headcount",
                        var_wei = "biomass",
                        var_surf = "surface")
```

```
# Add growth rates for gudgeon guild
mcmc.out <- mod_popgrow(df,
                        var_id = "pop_id",
                        var_tmp = "year",
                        var_tax = "taxa",
                        var_cnt = "headcount",
                        var_wei = "biomass",
                        var_surf = "surface",
                        var_guild = "gudgeon")


# Add growth rates at basin level
mcmc.out <- mod_popgrow(df,
                        var_id = "pop_id",
                        var_tmp = "year",
                        var_tax = "taxa",
                        var_cnt = "headcount",
                        var_wei = "biomass",
                        var_surf = "surface",
                        var_guild = "gudgeon",
                        var_reg = "hydro_basin")


# Add intermediate growth rates
mcmc.out <- mod_popgrow(df,
                        var_id = "pop_id",
                        var_tmp = "year",
                        var_tax = "taxa",
                        var_cnt = "headcount",
                        var_wei = "biomass",
                        var_surf = "surface",
                        var_guild = "gudgeon",
                        var_reg = "hydro_basin",
                        period = list(c(2010,2015),c(2015,2020)))


## End(Not run)
```

modenv_popgrow

Description

Fits Bayesian hierarchical single or multi-species population growth models with environmental stochasticity to abundance and/or biomass time series. The modelling framework not accounts for detection biases of taxa and individuals.

Usage

```
modenv_popgrow(
    df,
    var_id,
    var_tmp,
    var_env,
    var_tax = NULL,
    var_cnt = NULL,
    var_wei = NULL,
    var_surf = NULL,
    var_reg = NULL,
    var_guild = NULL,
    period = NULL,
    timestep = 1,
    save_parameters = NULL,
    n_chain = 3,
    n_iter = 10000,
    n_thin = n_iter/100,
    n_burnin = n_iter/4
)
```

Arguments

df	data.frame with variables used in the model
var_id	[character] variable name with sampling location identifiers (must not contain missing values)
var_tmp	[character] variable name with continuous sampling times (must be numeric without missing values)
var_env	[character] environmental covariate(s) name(s) whose values must be supplied by location and/or time (must be numeric without missing values)
var_tax	[optional, character] variable name with taxa identifiers to be specified for the multi-taxa models (must not contain missing values)
var_cnt	[optional, character] variable name with taxa abundances (must be numeric)

var_wei   [optional, character] variable name with taxa biomasses
(must be numeric)

var_surf   [optional, character] variable name with quantitative sampling effort
(e.g. sampling surfaces). If supplied, the model takes into account
variabilities in sampling effort from one time to the next
(must be numeric without missing values)

var_reg   [optional, character] variable name(s) with spatial level identifiers.
If supplied, estimates of occupancy changes rates at intermediate
spatial levels are added to the model. Otherwise, occupancy change
rates are only estimated across all locations

var_guild   [optional, character] variable name(s) with taxa guilds identifiers. If
supplied, estimates of guild occupancy changes rates are added to
the model

period   [optional, numeric list] list with start and end times for estimating
intermediate average growth rates

timestep   [optional, numeric] interval between two consecutive times to account
for regular gaps in time series in the rate estimates

save_parameters   [optional, character] name(s) of the parameter(s) to save (by default,
all model parametes are saved)

n_chain   [optional, numeric] number of MCMC chains

n_iter   [optional, numeric] number of iterations per MCMC chain

n_thin   [optional, numeric] backup interval for MCMC samples

n_burnin   [optional, numeric] number of initial iterations to discard per Markov
chain

Details

modoenv_popgrow relies on Bayesian hierarchical state-space models that include two or
three main components. The first component describes the occupancy dynamics of taxa at
locations defined by var_id from taxon- and location-specific probabilities of persistence
and colonisation. Two other independent components describe the population growth of
taxa on successively occupied locations, one based on abundance (if var_cnt is supplied)
and other based on biomass (if var_wei is supplied). These two components are based on
an autoregressive formulation in which growth rates are modelled as a linear function of
environmental covariates. At location, global (i.e. across all locations) and intermediate
spatial levels defined by var_reg, the finite population growth rates are estimated as the
changes in population size from one time to the next. Their geometric means are used
as average growth rates over the whole time modelling period and over intermediate time

periods defined by `period`. If `var_guild` is supplied, the guild occupancy growth rates are assessed from the geometric means of taxa growth rates.

The models are run using the `runMCMC` function of `nimble` package.

Value

`modenv_popgrow` returns a list with the summaries of main statistics associated with the estimation of the requested parameters (`mcmc_summary`), the MCMC chains (`mcmc_chain`) and the list of parameters subscripts (`subscript`).

See Also

`modenv_popgrowRS` and `get_modparametes`

Examples

```
## Not run:
data(riverfish)
df <- riverfish[riverfish$pass == 1,]

# Multi-taxa population growth from abundance
# with linear temperature effect on growth rates
mcmc.out <- modenv_popgrow(df,
                           var_id = "pop_id",
                           var_tmp = "year",
                           var_tax = "taxa",
                           var_env = "temperature",
                           var_cnt = "headcount")


# Account for variabilities in annual sampling surfaces
mcmc.out <- modenv_popgrow(df,
                           var_id = "pop_id",
                           var_tmp = "year",
                           var_tax = "taxa",
                           var_env = "temperature",
                           var_cnt = "headcount",
                           var_surf = "surface")


# Add growth rates for family and gudgeon guilds
mcmc.out <- modenv_popgrow(df,
                           var_id = "pop_id",
                           var_tmp = "year",
                           var_tax = "taxa",
                           var_env = "temperature",
                           var_cnt = "headcount",
                           var_surf = "surface",
                           var_guild = c("family","gudgeon"))
```

```
# Pike population growth from abundance and biomass
# with linear temperature effect on growth rates
mcmc.out <- modenv_popgrow(df[df$taxa == "pike",],
                           var_id = "pop_id",
                           var_tmp = "year",
                           var_env = "temperature",
                           var_cnt = "headcount",
                           var_wei = "biomass",
                           var_surf = "surface")


# Intermediate growth rates at basin level
mcmc.out <- modenv_popgrow(df[df$taxa == "pike",],
                           var_id = "pop_id",
                           var_tmp = "year",
                           var_env = "temperature",
                           var_cnt = "headcount",
                           var_wei = "biomass",
                           var_surf = "surface",
                           var_reg = "hydro_basin",
                           period = list(c(2010,2016),c(2016,2022)),
                           save_parameters = c("N_mulambda","B_mulambda"))

## End(Not run)
```

mod_popdyn

Description

Fits Bayesian hierarchical single or multi-taxa occupancy and growth models to abundance and/or biomass time series. The modelling framework not accounts for detection biases of taxa and individuals.

Usage

```
mod_popdyn(
    df,
    var_id,
    var_tmp,
    var_tax = NULL,
    var_cnt = NULL,
    var_wei = NULL,
    var_surf = NULL,
    var_reg = NULL,
    var_guild = NULL,
    period = NULL,
    timestep = 1,
    save_parameters = NULL,
    n_chain = 3,
    n_iter = 10000,
    n_thin = n_iter/100,
    n_burnin = n_iter/4
)
```

Arguments

df	data.frame with variables used in the model
var_id	[character] variable name with sampling location identifiers (must not contain missing values)
var_tmp	[character] variable name with continuous sampling times (must be numeric without missing values)
var_tax	[optional, character] variable name with taxa identifiers to be specified for the multi-taxa models (must not contain missing values)
var_cnt	[optional, character] variable name with taxa abundances (must be numeric)
var_wei	[optional, character] variable name with taxa biomasses (must be numeric)
var_surf	[optional, character] variable name with quantitative sampling effort (e.g. sampling surfaces). If supplied, the model takes into account

variabilities in sampling effort from one time to the next
(must be numeric without missing values)

var_reg	[optional, character] variable name(s) with spatial level identifiers. If supplied, estimates of occupancy changes rates at intermediate spatial levels are added to the model. Otherwise, occupancy change rates are only estimated across all locations
var_guild	[optional, character] variable name(s) with taxa guilds identifiers. If supplied, estimates of guild occupancy changes rates are added to the model
period	[optional, numeric list] list with start and end times for estimating intermediate average growth rates
timestep	[optional, numeric] interval between two consecutive times to account for regular gaps in time series in the rate estimates
save_parameters	[optional, character] name(s) of the parameter(s) to save (by default, all model parametes are saved)
n_chain	[optional, numeric] number of MCMC chains
n_iter	[optional, numeric] number of iterations per MCMC chain
n_thin	[optional, numeric] backup interval for MCMC samples
n_burnin	[optional, numeric] number of initial iterations to discard per Markov chain

Details

mod_popdyn relies on Bayesian hierarchical state-space models that include two or three main components. The first component describes the occupancy dynamics of taxa from presence-absence data (defined from abundances if var_cnt is supplied or, failing that, from biomasses). The successive occupancy states of locations defined by var_id depend on taxon- and location-specific probabilities of persistence and colonisation. At the global level (i.e. across all locations) and at intermediate spatial levels defined by var_reg, the taxa occupancy changes rates are estimated from the ratio of the number of occuped location from one time to the next. Their geometric mean is used as average occupancy rate over the whole time modelling period and over intermediate time periods defined by period. If var_guild is supplied, the guild occupancy change rates are assessed from the geometric means of taxa occupancy rates.

The other two components describe the population growth of taxa on successively occupied locations, one based on abundance (if var_cnt is supplied) and other based on biomass (if var_wei is supplied). These two components are based on an autoregressive growth model. At location, global (i.e. across all locations) and intermediate spatial levels defined by

var_reg, the finite population growth rates are estimated as the changes in population size from one time to the next. Their geometric means are used as average growth rates over the whole time modelling period and over intermediate time periods defined by period. If var_guild is supplied, the guild occupancy growth rates are assessed from the geometric means of taxa growth rates.

The models are run using the runMCMC function of nimble package.

Value

mod_popdyn returns a list with the summaries of main statistics associated with the estimation of the requested parameters (mcmc_summary), the MCMC chains (mcmc_chain) and the list of parameters subscripts (subscript).

See Also

mod_popoccup, mod_popgrow, mod_popdynRS and get_modparametes

Examples

```
## Not run:
data(riverfish)
df <- riverfish[riverfish$pass == 1,]

# Multi-taxa occupancy and population growth from abundance
mcmc.out <- mod_popdyn(df,
                       var_id = "pop_id",
                       var_tmp = "year",
                       var_tax = "taxa",
                       var_cnt = "headcount")

# Multi-taxa occupancy and population growth from biomass
mcmc.out <- mod_popdyn(df,
                       var_id = "pop_id",
                       var_tmp = "year",
                       var_tax = "taxa",
                       var_wei = "biomass")

# Multi-taxa occupancy and population growth from abundance and biomass
mcmc.out <- mod_popdyn(df,
                       var_id = "pop_id",
                       var_tmp = "year",
                       var_tax = "taxa",
                       var_cnt = "headcount",
                       var_wei = "biomass")
```

```
# Account for variabilities in annual sampling surfaces
mcmc.out <- mod_popdyn(df,
                       var_id = "pop_id",
                       var_tmp = "year",
                       var_tax = "taxa",
                       var_cnt = "headcount",
                       var_wei = "biomass",
                       var_surf = "surface")

# Add growth rates for family and gudgeon guilds at basin level
mcmc.out <- mod_popdyn(df,
                       var_id = "pop_id",
                       var_tmp = "year",
                       var_tax = "taxa",
                       var_cnt = "headcount",
                       var_wei = "biomass",
                       var_surf = "surface",
                       var_reg = "hydro_basin",
                       var_guild = c("family","gudgeon"))

# Pike occupancy and population growth from abundance and biomass
mcmc.out <- mod_popdyn(df[df$taxa == "pike",],
                       var_id = "pop_id",
                       var_tmp = "year",
                       var_cnt = "headcount",
                       var_wei = "biomass",
                       var_surf = "surface")

# Add rates for different time periods
mcmc.out <- mod_popdyn(df[df$taxa == "pike",],
                       var_id = "pop_id",
                       var_tmp = "year",
                       var_cnt = "headcount",
                       var_wei = "biomass",
                       var_surf = "surface",
                       period = list(c(2010,2014),c(2010,2018)))


## End(Not run)
```

modenv_popdyn

Description

Fits Bayesian hierarchical single or multi-taxa occupancy and growth models with environmental stochasticity to abundance and/or biomass time series. The modelling framework not accounts for detection biases of taxa and individuals.

Usage

```
modenv_popgrow(
    df,
    var_id,
    var_tmp,
    var_tax = NULL,
    var_cnt = NULL,
    var_wei = NULL,
    var_env = NULL,
    var_env0 = NULL,
    var_envP = NULL,
    var_envC = NULL,
    var_surf = NULL,
    var_reg = NULL,
    var_guild = NULL,
    period = NULL,
    timestep = 1,
    save_parameters = NULL,
    n_chain = 3,
    n_iter = 10000,
    n_thin = n_iter/100,
    n_burnin = n_iter/4
)
```

Arguments

`df`	data.frame with variables used in the model
`var_id`	[character] variable name with sampling location identifiers (must not contain missing values)
`var_tmp`	[character] variable name with continuous sampling times (must be numeric without missing values)
`var_tax`	[optional, character] variable name with taxa identifiers to be specified for the multi-taxa models (must not contain missing values)
`var_cnt`	[optional, character] variable name with taxa abundances (must be numeric)

var_wei	[optional, character] variable name with taxa biomasses (must be numeric)
var_env	[optional, character] environmental covariate(s) name(s) related to growth rates whose values must be supplied by location and/or time (must be numeric without missing values)
var_env0	[optional, character] environmental covariate(s) name(s) related to occupancy probabilities whose values must be supplied by location and/or by time (must be numeric without missing values)
var_envP	[optional, character] environmental covariate(s) name(s) related to persistence probabilities whose values must be supplied by location and/or by time (must be numeric without missing values)
var_envC	[optional, character] environmental covariate(s) name(s) related to colonisation probabilities whose values must be supplied by location and/or by time (must be numeric without missing values)
var_surf	[optional, character] variable name with quantitative sampling effort (e.g. sampling surfaces). If supplied, the model takes into account variabilities in sampling effort from one time to the next (must be numeric without missing values)
var_reg	[optional, character] variable name(s) with spatial level identifiers. If supplied, estimates of occupancy changes rates at intermediate spatial levels are added to the model. Otherwise, occupancy change rates are only estimated across all locations
var_guild	[optional, character] variable name(s) with taxa guilds identifiers. If supplied, estimates of guild occupancy changes rates are added to the model
period	[optional, numeric list] list with start and end times for estimating intermediate average growth rates
timestep	[optional, numeric] interval between two consecutive times to account for regular gaps in time series in the rate estimates
save_parameters	[optional, character] name(s) of the parameter(s) to save (by default, all model parametes are saved)
n_chain	[optional, numeric] number of MCMC chains
n_iter	[optional, numeric] number of iterations per MCMC chain
n_thin	[optional, numeric] backup interval for MCMC samples

n_burnin   [optional, numeric] number of initial iterations to discard per Markov
                    chain

Details

modenv_popdyn relies on Bayesian hierarchical state-space models that include two or three
main components. The first component describes the occupancy dynamics of taxa from
presence-absence data (defined from abundances if var_cnt is supplied or, failing that,
from biomasses). The successive occupancy states of locations defined by var_id depend
taxon-specific responses to environmental covariates if var_envO, var_envP or var_envC are
supplied (c.f. modenv_popoccup). Otherwise the successive occupancy states of locations
depend on taxon- and location-specific probabilities of persistence and colonisation. At the
global level (i.e. across all locations) and at intermediate spatial levels defined by var_reg,
the taxa occupancy changes rates are estimated from the ratio of the number of occuped
location from one time to the next. Their geometric mean is used as average occupancy
rate over the whole time modelling period and over intermediate time periods defined by
period. If var_guild is supplied, the guild occupancy change rates are assessed from the
geometric means of taxa occupancy rates.

The other two components describe the population growth of taxa on successively occupied
locations, one based on abundance (if var_cnt is supplied) and other based on biomass
(if var_wei is supplied). These two components are based on an autoregressive growth
formulation. If var_env is supplied the growth rates are modelled as a linear function of
environmental covariates. At location, global (i.e. across all locations) and intermediate
spatial levels defined by var_reg, the finite population growth rates are estimated as the
changes in population size from one time to the next. Their geometric means are used
as average growth rates over the whole time modelling period and over intermediate time
periods defined by period. If var_guild is supplied, the guild occupancy growth rates are
assessed from the geometric means of taxa growth rates.

The models are run using the runMCMC function of nimble package.

Value

modenv_popdyn returns a list with the summaries of main statistics associated with the
estimation of the requested parameters (mcmc_summary), the MCMC chains (mcmc_chain)
and the list of parameters subscripts (subscript).

See Also

modenv_popoccup, modenv_popgrow, modenv_popgdynRS and get_modparametes

Examples

```
## Not run:
data(riverfish)
df <- riverfish[riverfish$pass == 1,]
```

```
# Pike population dynamics
# with linear temperature effect on growth rates
mcmc.out <- modenv_popdyn(df[df$taxa == "pike",],
                          var_id = "pop_id",
                          var_tmp = "year",
                          var_cnt = "headcount",
                          var_env = "temperature",
                          var_surf = "surface")

# with linear temperature effect on occupancy probability
mcmc.out <- modenv_popdyn(df[df$taxa == "pike",],
                          var_id = "pop_id",
                          var_tmp = "year",
                          var_cnt = "headcount",
                          var_env0 = "temperature",
                          var_surf = "surface")

# with linear temperature effect on growth rates and occupancy probability
mcmc.out <- modenv_popdyn(df[df$taxa == "pike",],
                          var_id = "pop_id",
                          var_tmp = "year",
                          var_cnt = "headcount",
                          var_env = "temperature",
                          var_env0 = "temperature",
                          var_surf = "surface")

# Population dynamics of gudgeon species
# with environmental effects on persistence and colonisation probabilities
mcmc.out <- modenv_popdyn(df[!is.na(df$gudgeon),],
                          var_id = "pop_id",
                          var_tmp = "year",
                          var_tax = "taxa",
                          var_cnt = "headcount",
                          var_wei = "biomass",
                          var_envP = c("temperature","depth"),
                          var_envC = "temperature",
                          var_surf = "surface",
                          var_guild = "gudgeon")

## End(Not run)
```

mef_convertquali

Description

Converts one or more quantitative variables into qualitative variables.

Usage

```
mef_convertquali(df, var_quali)
```

Arguments

df   data frame

var_quali   [character] name(s) of qualitative variable(s) to convert

Details

mef_convertquali converts each quantitative variable by breaking it into multiple binary variables (one fewer than the number of modalities of qualitative variable).

Value

mef_convertquali returns the data frame df with the new binary variables.

Examples

```
data(riverfish)

# Conversion of variable flow
df <- mef_convertquali(riverfish, var_quali="flow")
```

mef_imputevalue

Description

Imputes missing values of a variable from observed values.

Usage

```
mef_imputevalue(df, var_id, var_tmp, var_imp)
```

Arguments

df   data frame

var_id   variable name with sampling location identifiers

var_tmp   variable name with continuous sampling times

var_imp   variable name with missing values to be imputed

Details

`mef_imputevalue` imputes missing values of a variable from observed values in order to get complete time series. At the location scale, the missing values are substituted by the previous time value or, failing that, by the following time value.

Value

`mef_imputevalue` returns the data frame `df` after imputing the missing values.

Examples

```
data(riverfish)
n <- sample(2010:2022, 4)
riverfish[n,"surface"] <- NA

# Imputation of surface missing values
df <- mef_imputevalue(riverfish, var_id=pop_id, var_tmp=year, var_imp=surface)
```

3. Model Description

3.1. Occupancy models

Occupancy models rely on hierarchical Bayesian state-space models that describe the occupancy dynamics of one or more taxa at level of sampling locations and allow assessment of change rates in taxa occupancy at various spatial levels and over different time periods.

3.1.1. Modelling occupancy from local extinction and colonisation events

In `mod_popoccup` and `mod_popoccupRS`, the local occupancy dynamics results from extinction, colonisation or recolonisation events. Let $z_{s,i,t}$, the true occupancy state of location i at time t by taxon s, this is given by the following formulation which helps to impute missing values in the location occupancy states:

$$z_{s,i,t} \sim Bernoulli\left(\phi_{s,i} \times z_{s,i,t-1} + \gamma_{s,i} \times (1 - z_{s,i,t-1})\right)$$

Thus, an occupied location at time $t-1$ remains occupied in the subsequent time with the taxon- and location-specific peristence probability $\phi_{s,i}$, whereas an unoccupied location at time $t-1$ is colonised or recolonised in the subsequent time with taxon- and location-specific colonisation probability $\gamma_{s,i}$ (Royle and Kéry, 2007; Chandler et al., 2015).
The taxon- and location-specific peristence probability is expressed as a logit function of taxon-specific persistence probability, ϕ_s, and a location-specific effect, $\epsilon_{\phi s,i}$:

$$logit(\phi_{s,i}) = \phi_s + \epsilon_{\phi s,i} \text{ with } \epsilon_{\phi s,i} \sim \mathcal{N}(0, \sigma^2_{\phi s,i})$$

The taxon- and location-specific colonisation probability is expressed as a logit function of taxon-specific colonisation probability, γ_s, and a location-specific effect, $\epsilon_{\gamma s,i}$:

$$logit(\gamma_{s,i}) = \gamma_s + \epsilon_{\gamma s,i} \text{ with } \epsilon_{\gamma s,i} \sim \mathcal{N}(0, \sigma^2_{\gamma s,i})$$

The taxon- and location-specific extinction probabilities are derived from persistence probabilities:

$$\xi_{s,i} = 1 - \phi_{s,i} \qquad\qquad \xi_s = 1 - \phi_s$$

3.1.2. Modelling occupancy from environmental covariates

In `modenv_popoccup` and `modenv_popoccupRS`, the local occupancy dynamics depend on taxa occurrence relationships with environmental covariates. Let us denote $z_{s,i,t}$, the true occupancy state of location i at time t by taxon s and $V_{k,i,t}$, k environmental covariates whose values vary over space and/or time (least one covariate).

If `var_env0` is supplied, the occupancy state depends on occupancy probability modelled as a linear function of environmental covariables (Welsh et al., 2013; Rota et al., 2016):

$$logit(\psi_{s,i,t}) = \alpha_s + \Sigma_k(\beta_{s,k} \times V_{k,i,t})$$
$$z_{s,i,t} \sim Bernoulli(\psi_{s,i,t})$$

where $\psi_{s,i,t}$ is the occupancy probability of location i at time t by taxon s, α_s is the taxon-specific intercept and $\beta_{s,k}$ are the taxon-specific and covariate-specific coefficients.

If `var_envP` is supplied, the occupancy state depends on peristence probability modelled as a linear function of environmental covariables:

$$logit(\phi_{s,i,t}) = \alpha_{\phi s} + \Sigma_k(\beta_{\phi s,k} \times V_{k,i,t})$$
$$z_{s,i,t} \sim Bernoulli\left(\phi_{s,i,t} \times z_{s,i,t-1} + \gamma_{s,i} \times (1 - z_{s,i,t-1})\right)$$

where $\phi_{s,i,t}$ is the peristence probability of location i at time t by taxon s, $\alpha_{\phi s}$ is the taxon-specific intercept, $\beta_{\phi s,k}$ are the taxon-specific and covariate-specific coefficients and $\gamma_{s,i}$ is the taxon- and location-specific colonisation probability (c.f. section 3.1.1).

If `var_envC` is supplied, the occupancy state depends on colonisation probability modelled as a linear function of environmental covariables:

$$logit(\gamma_{s,i,t}) = \alpha_{\gamma s} + \Sigma_k(\beta_{\gamma s,k} \times V_{k,i,t})$$
$$z_{s,i,t} \sim Bernoulli\left(\phi_{s,i} \times z_{s,i,t-1} + \gamma_{s,i,t} \times (1 - z_{s,i,t-1})\right)$$

where $\gamma_{s,i,t}$ is the colonisation probability of location i at time t by taxon s, $\alpha_{\gamma s}$ is the taxon-specific intercept, $\beta_{\gamma s,k}$ are the taxon-specific and covariate-specific coefficients and $\phi_{s,i}$ is the taxon- and location-specific perisistence probability (c.f. section 3.1.1).

If `var_envP` and `var_envC` are supplied, the occupancy state depends on both persistence and colonisation probabilities modelled as a linear function of environmental covariables:

$$logit(\phi_{s,i,t}) = \alpha_{\phi s} + \Sigma_k(\beta_{\phi s,k} \times V_{\phi\ k,i,t})$$
$$logit(\gamma_{s,i,t}) = \alpha_{\gamma s} + \Sigma_k(\beta_{\gamma s,k} \times V_{\gamma\ k,i,t})$$
$$z_{s,i,t} \sim Bernoulli\left(\phi_{s,i,t} \times z_{s,i,t-1} + \gamma_{s,i,t} \times (1 - z_{s,i,t-1})\right)$$

where $V_{\phi\ k,i,t}$ and $V_{\gamma\ k,i,t}$ are the environmental covariates respectively related to the persistence probability and the colonisation probability which may be the same or different.

3.1.3. Modelling detection probability by multi-pass sampling

In `mod_popoccupRS` and `modenv_popoccupRS`, the models account for an imperfect taxa detection to estimate the true occupancy states. This is done by estimating a taxon-specific detection probability using a multi-pass sampling method. The multi-pass sampling method is based on repeated population sampling over a short time period using the same protocol and the sampling effort. The samples form each pass can be removed or not. This method relies on the assumptions of population closure (i.e. no immigration, emigration, death or recruitment) and constant detection probability between pass (Rivot et al., 2008; Hedger et al., 2013).

Let $z_{s,i,t}$, the true occupancy state of location i at time t by taxon s, $x_{s,i,t,n}$ the observed occupancy state at pass n and π_s, the taxon-specific detection probability:

$$x_{s,i,t,n} \sim Bernoulli(\pi_s \times z_{s,i,t})$$

If `var_pro` is supplied, the models account for sampling protocol in estimating of detection probability:

$$x_{s,i,t,n} \sim Bernoulli(\pi_{s,p_{i,t}} \times z_{s,i,t})$$

where $\pi_{s,p_{i,t}}$ is the taxon-specific detection probability for sampling protocol p used at location i and time t.

If `var_det` is supplied, the taxon-specific detection probability is modelled as a linear function of k covariates, $V_{\pi\ k,i,t}$, whose values vary across locations and/or over time (least one covariate ; Welsh et al., 2013; Mollenhauer et al., 2022):

$$logit(\pi_{s,i,t}) = \alpha_{\pi s} + \Sigma_k(\beta_{\pi s,k} \times V_{\pi\ k,i,t})$$
$$x_{s,i,t,n} \sim Bernoulli(\pi_{s,i,t} \times z_{s,i,t})$$

where $\alpha_{\pi s}$ is the taxon-specific intercept and $\beta_{\pi s,k}$ are the taxon-specific and covariate-specific coefficients.

If `var_pro` and `var_det` are supplied, the taxon- and protocol-specific detection probability is modelled as a linear function of k covariates, $V_{\pi\ k,i,t}$ whose values vary across locations and/or over time:

$$logit(\pi_{s,i,t}) = \alpha_{\pi s,p_{i,t}} + \Sigma_k(\beta_{\pi s,k,p_{i,t}} \times V_{\pi\ k,i,t})$$
$$x_{s,i,t,n} \sim Bernoulli(\pi_{s,i,t} \times z_{s,i,t})$$

where $\alpha_{\pi s,p_{i,t}}$ is the taxon- and protocol-specific intercept and $\beta_{\pi s,k,p_{i,t}}$ are the taxon-, protocol- and covariate-specific coefficients.

3.1.4. Taxa occupancy change rates

At spatial level j, gobal level (i.e. across all locations) or intermediate spatial level (when `var_reg` is supplied), the occupancy change rates of taxon s at times t ($\lambda_{Zs,j,t}$) are defined from the ratio of the number of occupied locations by this taxon in level j from one time to the next. Then, the geometric mean of this occupancy change rates over the time period T (whole time modelling period and intermediate time periods when `period` is supplied) is used as average occupancy rate.

$$\lambda_{Zs,j,t} = \frac{\Sigma_i z_{s,j,i,t}}{\Sigma_i z_{s,j,i,t-1}} \qquad\qquad \bar{\lambda}_{Zs,j,T} = \left(\prod_{T_{\text{start}}}^{T_{\text{end}}} \lambda_{Zs,j,t}\right)^{1/(T_{\text{end}}-T_{\text{start}})}$$

The percentage of average occupancy changes of taxon s at level j over the time period T is derived from the average occupancy change rate:

$$R_{Zs,j,T} = 100 \times (\bar{\lambda}_{Zs,j,T} - 1)$$

The turnover rate of taxon s at level j and time t is also assess:

$$\tau_{s,j,t} = \frac{\Sigma_i z_{s,j,i,t} \times (1 - z_{s,j,i,t-1})}{\Sigma_i z_{s,j,i,t}}$$

The turnover rate is an indicator of the occupancy stability of taxon s in level j and can be interpreted as the probability that an occupied location i in level j is a newly occupied location (Royle and Kéry, 2007).

3.1.5. Guilds occupancy change rates

At spatial level j, gobal level (i.e. across all locations) or intermediate spatial level (when `var_reg` is supplied), the occupancy change rates of guild g at times t ($\lambda_{Zg,j,t}$) are defined from the mean geometric of the occupancy change rates of ns taxa s included in guild g and recorded in level j at time t. Then, the geometric mean of the occupancy change rates of guild g (($\lambda_{Zg,j,t}$) over the time period T (whole time modelling period and intermediate time periods when `period` is supplied) is used as average occupancy rate of guild g.

$$\lambda_{Zg,j,t} = \left(\prod_s \lambda_{Zg,s,j,t}\right)^{(1/ns)} \qquad \bar{\lambda}_{Zg,j,T} = \left(\prod_{T_{\text{start}}}^{T_{\text{end}}} \lambda_{Zg,j,t}\right)^{1/(T_{\text{end}} - T_{\text{start}})}$$

The percentage of average occupancy changes of guild g at level j over the time period T is derived from the average occupancy change rate:

$$R_{Zg,j,T} = 100 \times (\bar{\lambda}_{Zg,j,T} - 1)$$

3.1.6. Prior distributions

Parameter	Prior distributions
ϕ_s	$\mathcal{U}(0,1)$
$\sigma_{\phi s,i}$	$\Gamma(0.1, 0.1)$
γ_s	$\mathcal{U}(0,1)$
$\sigma_{\gamma s,i}$	$\Gamma(0.1, 0.1)$
α_s	$\mathcal{N}(0, \sigma_s^2)$ with $\sigma_s \sim \Gamma(0.1, 0.1)$
$\beta_{s,k}$	$\mathcal{N}(0, \sigma_{s,k}^2)$ with $\sigma_{s,k} \sim \Gamma(0.1, 0.1)$
$\alpha_{\phi s}$	$\mathcal{N}(0, \sigma_s^2)$ with $\sigma_s \sim \Gamma(0.1, 0.1)$
$\beta_{\phi s,k}$	$\mathcal{N}(0, \sigma_{s,k}^2)$ with $\sigma_{s,k} \sim \Gamma(0.1, 0.1)$
$\alpha_{\gamma s}$	$\mathcal{N}(0, \sigma_s^2)$ with $\sigma_s \sim \Gamma(0.1, 0.1)$
$\beta_{\gamma s,k}$	$\mathcal{N}(0, \sigma_{s,k}^2)$ with $\sigma_{s,k} \sim \Gamma(0.1, 0.1)$
π_s	$\mathcal{U}(0,1)$
$\pi_{s,p}$	$\mathcal{U}(0,1)$
$\alpha_{\pi s}$	$\mathcal{N}(0, \sigma_s^2)$ with $\sigma_s \sim \Gamma(0.1, 0.1)$
$\beta_{\pi s,k}$	$\mathcal{N}(0, \sigma_{s,k}^2)$ with $\sigma_{s,k} \sim \Gamma(0.1, 0.1)$
$\alpha_{\pi s,p}$	$\mathcal{N}(0, \sigma_{s,p}^2)$ with $\sigma_{s,p} \sim \Gamma(0.1, 0.1)$
$\beta_{\pi s,k,p}$	$\mathcal{N}(0, \sigma_{s,k,p}^2)$ with $\sigma_{s,k,p} \sim \Gamma(0.1, 0.1)$

4. References

Chandler, R. B., Muths, E., Sigafus, B. H., Schwalbe, C. R., Jarchow, C. J., and Hossack, B. R. (2015). Spatial occupancy models for predicting metapopulation dynamics and viability following reintroduction. *Journal of Applied Ecology*, 52(5):1325–1333.

Hedger, R. D., de Eyto, E., Dillane, M., Diserud, O. H., Hindar, K., McGinnity, P., Poole, R., and Rogan, G. (2013). Improving abundance estimates from electrofishing removal sampling. *Fisheries Research*, 137:104–115.

Mollenhauer, R., Brewer, S. K., Moore, D., Swedberg, D., Wedgeworth, M., Mollenhauer, R., Brewer, S. K., Moore, D., Swedberg, D., and Wedgeworth, M. (2022). A hierarchical approach to fish conservation in semiarid landscapes: A need to understand multiscale environmental relationships. In *River Basin Management - Under a Changing Climate*. IntechOpen.

Rivot, E., Prévost, E., Cuzol, A., Baglinière, J.-L., and Parent, E. (2008). Hierarchical Bayesian modelling with habitat and time covariates for estimating riverine fish population size by successive removal method. *Canadian Journal of Fisheries and Aquatic Sciences*, 65(1):117–133.

Rota, C. T., Ferreira, M. A. R., Kays, R. W., Forrester, T. D., Kalies, E. L., McShea, W. J., Parsons, A. W., and Millspaugh, J. J. (2016). A multispecies occupancy model for two or more interacting species. 7(10):1164–1173.

Royle, J. A. and Kéry, M. (2007). A Bayesian State-Space Formulation of Dynamic Occupancy Models. *Ecology*, 88(7):1813–1823.

Welsh, A. H., Lindenmayer, D. B., and Donnelly, C. F. (2013). Fitting and interpreting occupancy models. 8(1):e52015.