Package: moultmcmc (via r-universe)

October 26, 2024

Title Bayesian Inference for Avian Moult Phenology Models

Version 0.1.0

Date 2023-02-15

Description Precompiled Stan models to conduct full Bayesian inference in regression models for the phenology of avian primary moult based on the approach of Underhill & Zucchini (1988) <DOI:10.1111/j.1474-919X.1988.tb00993.x> and related hierarchical models to accommodate repeated-measures data (within-season recaptures of individuals).

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.1

Biarch true

Depends R (>= 3.4.0)

Imports methods, Rcpp (>= 0.12.0), RcppParallel (>= 5.0.1), rstan (>= 2.18.1), rstantools (>= 2.1.1), loo, tibble, stringr, dplyr, ggplot2, stats, magrittr, rlang, nlme, cowplot, tidyr, MASS, RColorBrewer, moult (>= 2.3.0), matrixStats

LinkingTo BH (>= 1.66.0), Rcpp (>= 0.12.0), RcppEigen (>= 0.3.3.3.0), RcppParallel (>= 5.0.1), rstan (>= 2.18.1), StanHeaders (>= 2.18.0)

Remotes cran/moult

SystemRequirements GNU make

Suggests testthat (>= 3.0.0), knitr, rmarkdown, remotes

Config/testthat/edition 3

VignetteBuilder knitr

Repository https://pboesu.r-universe.dev

RemoteUrl https://github.com/pboesu/moultmcmc

RemoteRef HEAD

RemoteSha 0a343e0589955fd917239590339acf7e73d6940d

Contents

`moultmcmc-package` *The 'moultmcmc' package.*

Description

Precompiled Stan models to conduct full Bayesian inference in regression models for the phenology of avian primary moult based on the approach of Underhill & Zucchini (1988) and related hierarchical models to accommodate repeated-measures data (within-season recaptures of individuals).

References

Boersch-Supan et al. (2022) Bayesian inference for models of moult duration and timing in birds arXiv DOI:10.48550/arXiv.2205.12120

Underhill & Zucchini (1988) A model for avian primary moult. Ibis 130:358 DOI:10.1111/j.1474-919X.1988.tb00993.x

Stan Development Team (2022). RStan: the R interface to Stan. R package version 2.21.5. https://mc-stan.org

---

`compare_plot` *Visual comparison of moult models*

Description

Visual comparison of moult models

Usage

```
compare_plot(..., names = NULL)
```

Arguments

`...`	two or a moult or moultmcmc model
`names`	optional character vector of model names

Value

a plot comparing parameter estimates and their uncertainties

`consolidate_moult_records` *Consolidate a mixture of continuous and categorical moult records*

Description

This is a helper function to format input data for the Type 1+2 moult model.

Usage

```
consolidate_moult_records(moult_score, moult_cat)
```

Arguments

`moult_score`	a numeric vector of (linearized) moult scores in [0,1] (0 = old plumage,1 = new plumage).
`moult_cat`	a numeric vector of categorical moult codes (1 = old plumage,2 = moulting, 3 = new plumage)

Value

a numeric vector of scores and categorical records with values [0,1] for old, new and continuous active moult scores, and value 2 for categorical active moult records.

`fixef.moultmcmc` *Extract Population-Level Estimates*

Description

Extract the population-level ('fixed') effects from a `moultmcmc` object.

Usage

```
## S3 method for class 'moultmcmc'
fixef(object, summary = TRUE, probs = c(0.025, 0.975), pars = NULL, ...)
```

Arguments

`object`	a moultmcmc model
`summary`	logical, should posterior samples be summarised for each parameter
`probs`	numeric, desired quantiles for summary statistics
`pars`	Optional names of coefficients to extract. By default, all coefficients are extracted.
`...`	Currently ignored.

Value

If `summary` is `TRUE`, a matrix for the population-level effects. If `summary` is `FALSE`, a matrix with one row per posterior draw and one column per population-level effect.

`loo.moultmcmc` *Efficient approximate leave-one-out cross-validation (LOO)*

Description

Efficient approximate leave-one-out cross-validation (LOO)

Usage

```
## S3 method for class 'moultmcmc'
loo(x, cores = getOption("mc.cores", 1))
```

Arguments

`x`	A moultmcmc model object
`cores`	The number of cores to use for parallelization. This defaults to the option mc.cores which can be set for an entire R session by options(mc.cores = NUMBER). The old option loo.cores is now deprecated but will be given precedence over mc.cores until loo.cores is removed in a future release. As of version 2.0.0 the default is now 1 core if mc.cores is not set, but we recommend using as many (or close to as many) cores as possible. Note for Windows 10 users: it is strongly recommended to avoid using the .Rprofile file to set mc.cores (using the cores argument or setting mc.cores interactively or in a script is fine).

Value

The loo() methods return a named list with class c("psis_loo", "loo"). See ?loo::loo

---

`moultmcmc` *Bayesian inference for Underhill-Zucchini moult models and expansions*

Description

Bayesian inference for Underhill-Zucchini moult models and expansions

Usage

```
moultmcmc(
  moult_column,
  date_column,
  id_column = NULL,
  start_formula = ~1,
  duration_formula = ~1,
  sigma_formula = ~1,
  type = 2,
  lump_non_moult = FALSE,
  data,
  init = "auto",
  flat_prior = TRUE,
  beta_sd = 0,
  log_lik = FALSE,
  use_phi_approx = FALSE,
  active_moult_recaps_only = TRUE,
  same_sigma = FALSE,
  ...
)
```

Arguments

`moult_column`	the name the column in `data` containing moult indices, i.e. a numeric vector of (linearized) moult scores in [0,1] (0 = old plumage, 1 = new plumage; for model types 1-5), numerical moult codes (1 = old plumage, 2 = moulting, 3 = new plumage; for model type 1), or a mixed column created by `consolidate_moult_records` for model type 12.
`date_column`	the name the column in `data` containing sampling dates, encoded as days since an arbitrary reference date, i.e. a numeric vector
`id_column`	(optional) factor identifier. Usually a season-individual combination to encode within-season recaptures, defaults to NULL. When provided moultmcmc will attempt to fit the relevant recaptures model.
`start_formula`	model formula for start date
`duration_formula`	model formula for duration
`sigma_formula`	model formula for start date sigma
`type`	integer (one of 1,2,3,4,5,12) referring to type of moult data and consequently model to be fitted (see details)
`lump_non_moult`	logical; should pre- and post-moult observations be treated as indistinguishable? if TRUE and type %in% c(1,2,12), the relevant lumped model will be fitted (see details).
`data`	Input data frame must contain a numeric column "date" and a column "moult_cat" which is a numeric vector of categorical moult codes (1 = old plumage,2 = moulting,3 = new plumage).
`init`	Specification of initial values for all or some parameters. Can be the string "auto" for an automatic guess based on the data, or any of the permitted `rstan` options: the digit 0, the strings "0" or "random", or a function. See the detailed documentation for the init argument in `rstan::stan`.
`flat_prior`	use uniform prior on start date and duration (TRUE) or vaguely informative truncated normal prior (FALSE). Defaults to TRUE.
`beta_sd`	use zero-centred normal priors for regression coefficients other than intercepts? If <= 0 the stan default of improper flat priors is used.
`log_lik`	boolean retain pointwise log-likelihood in output? This enables model assessment and selection via the loo package. Defaults to FALSE, can lead to very large output arrays when sample size is large.
`use_phi_approx`	logical flag whether to use stan's Phi_approx function to calculate the "old" likelihoods
`active_moult_recaps_only`	logical flag whether to ignore repeated observations outside the active moult phase
`same_sigma`	logical flag, currently unused
`...`	Arguments passed to `rstan::sampling` (e.g. iter, chains).

Details

type refers to the type of moult data available (see Underhill and Zucchini (1998) and Underhill, Zucchini and Summers (1990)).

type = 1 sample is representative of entire population (not yet moulting, in moult, and birds which have completed moult). For type 1 data, any value between 0 and 1 (> 0 and < 1) can be used as the moult index for birds in active moult. The value used does not matter, only the fact that they are in moult. type = 2 (default) sample is representative of entire population (not yet moulting, in moult, and birds which have completed moult). Moult scores are required.

type = 3 sample is representative only of birds in moult. Individuals with moult scores 0 or 1 are ignored.

type = 4 sample is representative only of birds in moult and those that have completed moult. Individuals with moult scores 0 are ignored.

type = 5 sample is representative only of birds that have not started moult or that are in moult. Individuals with moult scores 1 are ignored.

Value

An object of class `stanfit` returned by `rstan::sampling`

`moult_plot` *moult_plot generic*

Description

moult_plot generic

Usage

```
moult_plot(...)
```

Arguments

`...`	...

Value

...

`moult_plot.moult` *Moult plot for moult model*

Description

Moult plot for moult model

Usage

```
## S3 method for class 'moult'
moult_plot(x, prob = 0.95, plot_data = TRUE, plot = TRUE, ...)
```

Arguments

`x`	moult model object created with moult::moult
`prob`	coverage probability of active moult in the sampled population
`plot_data`	logical, if TRUE plot observations
`plot`	logical, if TRUE (default) return a plot, else return a dataframe with start and end dates of moult based on model intercepts
`...`	not currently used

Value

a plot or data.frame

`moult_plot.moultmcmc` *Moult plot for moultmcmc model*

Description

Moult plot for moultmcmc model

Usage

```
## S3 method for class 'moultmcmc'
moult_plot(
  x,
  prob = 0.95,
  prob_ci = NULL,
  plot_data = TRUE,
  plot = TRUE,
  newdata = NULL,
  col.data = NULL,
  ...
)
```

Arguments

`x`	moultmcmc model object
`prob`	coverage probability of active moult in the sampled population
`prob_ci`	coverage probability of credible interval
`plot_data`	logical, if TRUE plot observations
`plot`	logical, if TRUE (default) return a plot, else return a dataframe with start and end dates of moult based on model intercepts
`newdata`	optional data.frame of conditions to plot (passed to predict.moultmcmc)
`col.data`	optional, name of column in model data to colour data points
`...`	not currently used

Value

a plot or data.frame

`namedList` *internal helper function to create a named list*

Description

original version by Ben Bolker: https://stackoverflow.com/a/16951524

Usage

```
namedList(...)
```

Arguments

`...`	args

Value

a named list

`predict.moultmcmc` *Predict method for moultmcmc models*

Description

Predict method for moultmcmc models

Usage

```
## S3 method for class 'moultmcmc'
predict(
  object,
  newdata = NULL,
  predict.type = "parameters",
  summary = TRUE,
  intervals = 0.1,
  prob = 0.95,
  ...
)
```

Arguments

`object`	a fitted moultmcmc model
`newdata`	data.frame with explanatory variables for which to make predictions
`predict.type`	specifies form of predictions, see details.
`summary`	logical, if TRUE (default) return a data.frame of posterior means, otherwise return a list of arrays of the full posterior sample of the predicted quantity (with one list element per predicted quantity and array dimensions nrow(newdata) *number or posterior draws)
`intervals`	not currently used
`prob`	nominal coverage probability of credible interval
`...`	further arguments

Value

a data.frame or list, depending on input arguments

`ranef.moultmcmc` *Extract Individual-Level Estimates*

Description

Extract the individual-level ('random') effects from a `moultmcmc` object.

Usage

```
## S3 method for class 'moultmcmc'
ranef(object, summary = TRUE, probs = c(0.025, 0.975), pars = NULL, ...)
```

Arguments

`object`	a moultmcmc model
`summary`	logical, should posterior samples be summarised for each parameter
`probs`	numeric, desired quantiles for summary statistics
`pars`	Optional names of coefficients to extract. By default, all coefficients are extracted.
`...`	Currently ignored.

Value

If `summary` is `TRUE`, a matrix for the individual-level effects. If `summary` is `FALSE`, a matrix with one row per posterior draw and one column per individual-level effect.

`recaptures` *Simulated moult scores for 235 individuals with no heterogeneity in duration*

Description

A dataset containing the moult scores and capture dates for a simulated population of passerines including recaptures

Usage

```
recaptures
```

Format

A data frame with 299 rows and 13 variables:

- **id** individual identifier
- **sample_size** number of observations for the said individual
- **start_date** individual moult initiation date
- **duration** individual duration
- **set** simulation set
- **date_sampled** observation date (days since January 01)
- **pfmg_sampled** true PFMG at sampling occasion
- **pfmg_with_error** PFMG incorporating measurement error, if any
- **sampling_bias** descriptor of sampling bias type, if any
- **prop_recaptured_individuals** proportion of individuals with more than 1 capture
- **rel_tau_sd** relative population standard deviation of the moult duration
- **obs_error** measurement error
- **scenario** simulation scenario

Source

P. Boersch-Supan

`recaptures2` *Simulated moult scores for 235 individuals with heterogeneity in durations*

Description

A dataset containing the moult scores and capture dates for a simulated population of passerines including recaptures. Simulation is based on an underlying population moult start date of 196.83, moult duration of 77.80 days and a population standard deviation of the start date of 8.1 days.

Usage

```
recaptures2
```

Format

A data frame with 299 rows and 13 variables:

- **id** individual identifier
- **sample_size** number of observations for the said individual
- **start_date** individual moult initiation date
- **duration** individual duration

set	simulation set
date_sampled	observation date (days since January 01)
pfmg_sampled	true PFMG at sampling occasion
pfmg_with_error	PFMG incorporating measurement error, if any
sampling_bias	descriptor of sampling bias type, if any
prop_recaptured_individuals	proportion of individuals with more than 1 capture
rel_tau_sd	relative population standard deviation of the moult duration
obs_error	measurement error
scenario	simulation scenario

Source

P. Boersch-Supan

`residual_plot` *residual_plot generic*

Description

residual_plot generic

Usage

```
residual_plot(...)
```

Arguments

`...`	...

Value

...

`residual_plot.moult` *Residual plot for moult model*

Description

This function displays a plot that shows where observed dates and moult scores fall relative to the predictions of the fitted model. Active moult observations should generally fall within +/- 3 start-date standard deviations of the regression line.

Usage

```
## S3 method for class 'moult'
residual_plot(x, plot = TRUE, ...)
```

Arguments

`x`	moult model object created with moult::moult
`plot`	logical, if TRUE (default) return a plot, else return a data.frame with calculated quantities
`...`	not currently used

Value

a plot

`siskins` *Moult records of Eurasian Siskins*

Description

A dataset containing simulated moult scores and capture dates of Eurasian Siskins (Spinus spinus), based on a sub-urban Scottish population.

Usage

```
siskins
```

Format

A data frame with 299 rows and 3 variables:

yday sampling date (days since Jan 01)

pfmg PFMG at sampling occasion

id individual identifier

start_date "true" individual start date

duration "true" individual moult duration

Source

Philipp Boersch-Supan

References

Insley et al. (in prep). Breeding and moult phenology of siskins in the Scottish Highlands.

`summary.moultmcmc` *Summary method for moultmcmc models*

Description

Summary method for moultmcmc models

Usage

```
## S3 method for class 'moultmcmc'
summary(object, ...)
```

Arguments

`object`	a moultmcmc object
`...`	passed to summary_table

Value

a summary table

`summary_table` *Summary table generic*

Description

Summary table generic

Usage

```
summary_table(...)
```

Arguments

`...`	...

Value

...

---

summary_table.moult *Summary Table for moult model*

Description

Summary Table for moult model

Usage

```
## S3 method for class 'moult'
summary_table(x, prob = 0.95, tidy_names = TRUE, ...)
```

Arguments

`x`	moult model object created with moult::moult
`prob`	nominal coverage probability of confidence interval
`tidy_names`	adjust default parameter names from moult() to follow consistent nomenclature of model.matrix()
`...`	not currently used

Value

a tibble

Examples

```
## Not run: m1 <- moult::moult(Mindex ~ Day, data = sanderlings)
## Not run: summary_table(m1)
```

---

summary_table.moultmcmc *Summary Table*

Description

Summary Table

Usage

```
## S3 method for class 'moultmcmc'
summary_table(
  x,
  pars = x$stanfit@sim$pars_oi,
  prob = 0.95,
  include = TRUE,
  ...
)
```

Arguments

`x`	moultmcmc fit object
`pars`	A character vector of parameter names. The default is all parameters for which samples are saved. If include = FALSE, then the specified parameters are excluded from the printed summary.
`prob`	nominal coverage probability of credible interval
`include`	Logical scalar (defaulting to TRUE) indicating whether to include or exclude the parameters named by the pars argument.
`...`	Additional arguments passed to the summary method for stanfit objects.

Value

a tibble

`weavers_processed` *Moult records of Southern Masked Weavers*

Description

A dataset containing the moult scores and capture dates for Southern Masked Weavers (Ploceus velatus) from the Western Cape, South Africa.

Usage

```
weavers_processed
```

Format

A data frame with 5282 rows and 6 variables:

RDate sampling date (calendar date)

Sex SAFRING sex code

Year Sampling Year

Moult feather scores for primaries

pfmg Proportion of feather mass grown

day sampling date (days since August 01 of each year)

Source

Erni et al. 2013

References

Oschadleus 2005, Patterns of primary moult in the weavers, Ploceidae. University of Cape Town.

Index