Package 'bsam'

November 17, 2016

Type Package

Title Bayesian State-Space Models for Animal Movement

Version 1.1.1

Depends R (>= 3.3.0), rjags (>= 4-6)

Imports coda (>= 0.18-1), dplyr (>= 0.5.0), ggplot2 (>= 2.1.0), gridExtra (>= 2.2.1), msm (>= 1.6.1), mvtnorm (>= 1.0-5), rworldxtra(>= 1.01), sp (>= 1.2-3), tibble(>= 1.1)

SystemRequirements JAGS (>= 4.2.0)

Description

Tools to fit Bayesian state-space models to animal tracking data. Models are provided for location filtering, location filtering and behavioural state estimation, and their hierarchical versions. The models are primarily intended for fitting to ARGOS satellite tracking data but options exist to fit to other tracking data types. For Global Positioning System data, consider the 'moveHMM' package. Simplified Markov Chain Monte Carlo convergence diagnostic plotting is provided but users are encouraged to explore tools available in packages such as 'coda' and 'boa'.

License GPL-2

URL https://github.com/ianjonsen/bsam

BugReports https://github.com/ianjonsen/bsam/issues

LazyData yes

RoxygenNote 5.0.1

NeedsCompilation no

Author Ian Jonsen [aut, cre],
Sophie Bestley [ctb],
Simon Wotherspoon [ctb],
Michael Sumner [ctb]

Maintainer Ian Jonsen <ian.jonsen@mq.edu.au>

Repository CRAN

Date/Publication 2016-11-17 14:34:35

R topics documented:

`bsam-package` *Fit Bayesian state-space models to animal tracking data*

Description

Models provided are DCRW (for location filtering), DCRWS (for location filtering and behavioural state estimation), and their hierarchical versions (hDCRW, hDCRWS) to estimate parameters jointly across multiple individual tracking datasets. The models are fit in JAGS using Markov chain Monte Carlo simulation methods. The models are intended to be fit to Argos satellite tracking data but options exist to allow fits to other tracking data types (type `?fit_ssm` for details).

Details

Package:	bsam
Type:	Package
Version:	1.1.0
Date:	2016-11-01
License:	GPL-2
LazyLoad:	yes

Fit Bayesian state-space models to Argos satellite tracking data. Models provided are DCRW - for location filtering; DCRWS - for location filtering and behavioural state estimation with 2 behavioural states; hDCRW and hDCRWS - hierarchical models for location filtering only, and location filtering with behavioural state estimation, respectively, across multiple animals.

The hierarchical models may provide improved location and/or behavioural state estimates compared to fitting DCRW/DCRWS to individual datasets.

Author(s)

Ian Jonsen

Maintainer: Ian Jonsen <ian.jonsen@mq.edu.au>

References

Jonsen ID, Mills Flemming J, Myers RA (2005) Robust state-space modeling of animal movement data. Ecology 86:2874-2880

Jonsen ID (2016) Joint estimation over multiple individuals improves behavioural state inference from animal movement data. Scientific Reports 6:20625

See Also

fit_ssm

Examples

```
## Not run:
# Fit DCRW model for state filtering and regularization
data(ellie)
fit <- fit_ssm(ellie, model = "DCRW", tstep = 1, adapt = 5000, samples = 5000,
               thin = 5, span = 0.2)
diag_ssm(fit)
map_ssm(fit)
plot_fit(fit)

## End(Not run)
```

---

dat4jags *Correlated Random Walk Filter*

Description

Format track data for filtering

Usage

```
dat4jags(d, tstep = 1, tpar = tpar())
```

Arguments

d	a data frame of observations (see details)
tstep	the time step to predict to (in days)
tpar	generalised t-distribution parameters for ARGOS location classes. By default dat4jags uses the parameters estimated in Jonsen et al (2005) Ecology 86:2874-2880 but users may specify other ARGOS error parameter values via the tpar function.

Details

This is an internal function used by `fit_ssm` to format track data for JAGS.

The input track is given as a dataframe where each row is an observed location and columns

'id' individual animal identifier,

'date' observation time (POSIXct,GMT),

'lc' ARGOS location class,

'lon' observed longitude,

'lat' observed latitude.

Location classes can include Z, F, and G; where the latter two are used to designate fixed (known) locations (e.g. GPS locations) and "generic" locations (e.g. geolocation data) where the user supplies the error standard deviations, either via the `tpar` function or as two extra columns in the input data.

From this `dat4jags` calculates interpolation indices `idx` and weights `ws` such that if `x` is the matrix of predicted states, the fitted locations are `ws*x[idx+1,] + (1-ws)*x[idx+2,]`.

Value

A list with components

`id`	the unique identifier for each dataset
`y`	a 2 column matrix of the lon,lat observations
`itau2`	a 2 column matrix of the ARGOS precision (1/scale) parameters
`nu`	a 2 column matrix of the ARGOS df parameters
`idx`	a vector of interpolation indices
`ws`	a vector of interpolation weights
`ts`	the times at which states are predicted (POSIXct,GMT)
`obs`	the input observed data frame
`tstep`	the time step specified in the `fitSSM` call

References

Jonsen ID, Mills Flemming J, Myers RA (2005) Robust state-space modeling of animal movement data. Ecology 86:2874-2880 (Appendix A)

diag_ssm *Plot standard McMC convergence diagnostics to help determine lack of model convergence.*

Description

Takes a fitted `fit_ssm` object and uses standard McMC convergence diagnostic plots to aid assessment of lack of convergence.

Usage

```
diag_ssm(fit)
```

Arguments

`fit`	an output object from `fit_ssm`

Value

Uses plotting functions from Martyn Plummer's `coda` package to help diagnose lack of convergence for the core model parameters. The traceplot shows the time series for both McMC chains; the density plot shows the density estimate for each parameter; the autocorrelation plots show the within-chain sample autocorrelation for each parameter; the G-B-R shrink factor plot shows the evolution of Gelman and Rubin's shrink factor for increasing number of iterations. See the `coda` package for further details.

References

Brooks SP, Gelman A (1998) General methods for monitoring convergence of iterative simulations. Journal of Computational and Graphical Statistics 7:434-455

Examples

```
## Not run:
data(ellie)
fit.s <- fit_ssm(ellie, model = "DCRWS", tstep = 1, adapt = 2000, samples = 1000,
              thin = 2, span = 0.1)
diag_ssm(fit.s)

# increase burnin, posterior sample numbers, and thinning factor
fit.s2 <- fit_ssm(ellie, model = "DCRWS", tstep = 1, adapt = 5000, samples = 5000,
              thin = 5, span = 0.1)
diag_ssm(fit.s2)

## End(Not run)
```

ellie *Elephant seal Argos satellite data (2 individuals)*

Description

Example elephant seal Argos tracking data. Data were sourced from the Integrated Marine Observing System (IMOS) - IMOS is supported by the Australian Government through the National Collaborative Research Infrastructure Strategy and the Super Science Initiative.

Format

.RData

fitSSM *Deprecated functions.*

Description

`fitSSM`, `diagSSM`, and `plotSSM`, have been deprecated. Instead use `fit_ssm`, `diag_ssm` and `map_ssm`.

Usage

```
fitSSM(...)

diagSSM(...)

plotSSM(...)
```

Arguments

`...`	ignored

fit_ssm *Fit Bayesian state-space models to animal movement data*

Description

Fits state-space models to animal tracking data. User can choose between a first difference correlated random walk (DCRW) model, a switching model (DCRWS) for estimating location and behavioural states, and thier hierarchical versions (hDCRW, hDCRWS). The models are structured for Argos satellite data but options exist for fitting to other tracking data types.

Usage

```
fit_ssm(data, model = "DCRW", tstep = 1, adapt = 10000, samples = 5000,
  thin = 5, span = 0.2)
```

Arguments

data	A data frame containing the following columns, "id","date", "lc", "lon", "lat". "id" is a unique identifier for the tracking dataset. "date" is the GMT date-time of each observation with the following format "2001-11-13 07:59:59". "lc" is the Argos location quality class of each observation, values in ascending order of quality are "Z", "B", "A", "0", "1", "2", "3". "lon" is the observed longitude in decimal degrees. "lat" is the observed latitude in decimal degress. The Z-class locations are assumed to have the same error distributions as B-class locations. Optionally, the input data.frame can specify the error standard deviations for longitude and latitude (in units of degrees) in the last 2 columns, named "lonerr" and "laterr", respectively. These errors are assumed to be normally distributed. When specifying errors in the input data, all "lc" values must be equal to "G". This approach allows the models to be fit to data types other than Argos satellite data, e.g. geolocation data. See `dat4jags` for other options for specifying error parameters. WARNING: there is no guarantee that invoking these options will yield sensible results! For GPS data, similar models can be fit via the `moveHMM` package.
model	name of state-space model to be fit to data. This can be one of "DCRW", "DCRWS", "hDCRW", or "hDCRWS"
tstep	time step as fraction of a day, default is 1 (24 hours).
adapt	number of samples during the adaptation and update (burn-in) phase, adaptation and updates are fixed at adapt/2
samples	number of posterior samples to generate after burn-in
thin	amount of thining of to be applied to the posterior samples to minimize within-chain sample autocorrelation
span	parameter that controls the degree of smoothing by `stats::loess`, used to obtain initial values for the location states. Smaller values = less smoothing. Values > 0.2 may be required for sparse datasets

Details

The models are fit using JAGS 4.2.0 (Just Another Gibbs Sampler, created and maintained by Martyn Plummer; http://martynplummer.wordpress.com/; http://mcmc-jags.sourceforge.net). `fit_ssm` is a wrapper that first calls `dat4jags`, which prepares the input data, then calls `ssm` or `hssm`, which fit the specified state-space model to the data, returning a list of results.

Value

For DCRW and DCRWS models, a list is returned with each outer list elements corresponding to each unique individual id in the input data Within these outer elements are a "summary" data.frame of posterior mean and median state estimates (locations or locations and behavioural states), the

name of the "model" fit, the "timestep" used, the input location "data", the number of location state estimates ("N"), and the full set of "mcmc" samples. For the hDCRW and hDCRWS models, a list is returned where results, etc are combined amongst the individuals

Author(s)

Ian Jonsen

References

Jonsen ID, Mills Flemming J, Myers RA (2005) Robust state-space modeling of animal movement data. Ecology 86:2874-2880

Block et al. (2011) Tracking apex marine predator movements in a dynamic ocean. Nature 475:86-90

Jonsen et al. (2013) State-space models for biologgers: a methodological road map. Deep Sea Research II DOI: 10.1016/j.dsr2.2012.07.008

Jonsen (2016) Joint estimation over multiple individuals improves behavioural state inference from animal movement data. Scientific Reports 6:20625

Examples

```
## Not run:
# Fit DCRW model for state filtering and regularization
data(ellie)
fit <- fit_ssm(ellie, model = "DCRW", tstep = 2, adapt = 5000, samples = 5000,
              thin = 5, span = 0.2)
diag_ssm(fit)
map_ssm(fit)
plot_fit(fit)

# Fit DCRWS model for state filtering, regularization and behavioural state estimation
 fit.s <- fit_ssm(ellie, model = "DCRWS", tstep = 2, adapt = 5000, samples = 5000,
                thin = 5, span = 0.2)
 diag_ssm(fit.s)
 map_ssm(fit.s)
 plot_fit(fit.s)

# fit hDCRWS model to > 1 tracks simultaneously
# this may provide better parameter and behavioural state estimation
# by borrowing strength across multiple track datasets
 hfit.s <- fit_ssm(ellie, model = "hDCRWS", tstep = 2, adapt = 5000, samples = 5000,
                thin = 5, span = 0.2)
 diag_ssm(hfit.s)
 map_ssm(hfit.s)
 plot_fit(hfit.s)

## End(Not run)
```

get_summary *Extract summary output and optionally export as a .csv file.*

Description

Takes a fitted `fit_ssm` object and extracts the summary data.frame, which includes the animal ids, POSIXct date/time (at increments specified by `tstep` in the `fit_ssm` call), posterior mean longitude and latitude, and the 2.5, 50, and 97.5 longitude and latitude. For the `DCRWS` and `hDCRWS` models, the posterior mean and median behavioural states corresponding to each estimated location are also provided.

Usage

```
get_summary(x, file = " ")
```

Arguments

`x`	an output object from `fit_ssm`. If not an error will be returned.
`file`	a character string naming a file. " " indicates output to the console (default)

Value

a summary data.frame printed either to the console (default) or written as .csv to a specified file.

Examples

```
## Not run:
data(ellie)
fit <- fit_ssm(ellie, model = "DCRW", tstep = 1, adapt = 2000, samples = 1000,
               thin = 2, span = 0.1)

## print to console
get_summary(fit)

## export to .csv file
get_summary(fit, file = "ellie_dcrw.csv")

## plot track of second animal (posterior mean longitude and latitude)
p <- ggplot(data = subset(get_summary(fit), id == unique(id)[2])) +
       geom_point(aes(lat, lon), col = "firebrick", size = 2)
p

## overlay posterior median longitude and latitude
p + geom_point(aes(lat.5, lon.5), col = "dodgerblue", size = 1)

## End(Not run)
```

hssm *Fits hierarchical state-space models to Argos data*

Description

Takes output from `dat4jags`, sets up initial values, calls JAGS, and aggregates results. Intended for internal use, called by `fit_ssm`.

Usage

```
hssm(d, model = "hDCRWS", adapt, samples, thin, chains, span)
```

Arguments

`d`	structured data from `dat4jags` to be passed to JAGS
`model`	the state-space model to be fit: hDCRW or hDCRWS
`adapt`	number of samples in adaptation/burnin phase
`samples`	number of posterior samples
`thin`	thinning factor to reduce posterior sample autocorrelation
`chains`	number of parallel McMC chains to run
`span`	span

Value

Returns a list of McMC samples from marginal posteriors and a summary `data.frame` of mean and median position estimates.

See Also

Function to be called by `fit_ssm`.

map_ssm *Plot estimated track, behavioural states and observations on a map.*

Description

Takes a fitted `fit_ssm` object and plots the observed (data) and estimated locations on a map. For the behavioural models (DCRWS, hDCRWS), the estimated locations are coloured according to the posterior mean behavioural state estimates.

Usage

```
map_ssm(fit, onemap = TRUE)
```

Arguments

fit	an output object from fit_ssm
onemap	If TRUE (default) then all estimated tracks are plotted on a single, combined map, if FALSE then tracks are plotted on separate maps.

Value

Observed locations are plotted as '+' symbols and estimated locations as filled circles. Individual track id's (for DCRW and DCRWS models) are displayed at the top of each plot, but only when onemap = FALSE. The model specified in fit_ssm is also displayed at the top. Takes advantage of ggplot2 plotting functions.

Currently, results from the hierarchical models (hDCRW, hDCRWS) can only be plotted on a combined map.

Examples

```
## Not run:
data(ellie)
fit.s <- fitSSM(ellie, model = "DCRWS", tstep = 1, adapt = 2000, samples = 1000,
              thin = 2, span = 0.1)
map_ssm(fit.s, onemap = TRUE)

hfit.s <- fit_ssm(ellie, model = "hDCRWS", tstep = 1, adapt = 2000, samples = 1000,
                thin = 2, span = 0.1)
map_ssm(hfit.s)

## End(Not run)
```

---

plot_fit *Plot the 1-D time-series of estimated location and behavioural states*

Description

Takes a fitted fit_ssm object and plots the observed (data), estimated location and behavioural states (posterior means) as 1-D time-series. Each individual dataset is plotted separately.

Usage

```
plot_fit(fit)
```

Arguments

fit	an output object from fit_ssm

Value

Observed locations are plotted as filled circles and estimated locations as blue lines with the 95% credible interval as a ribbon. Uses ggplot2 plotting functions.

Examples

```
## Not run:
data(ellie)
fit.s <- fit_ssm(ellie, model = "DCRWS", tstep = 1, adapt = 2000, samples = 1000,
              thin = 2, span = 0.1)
plot_fit(fit.s)

hfit.s <- fit_ssm(ellie, model = "hDCRWS", tstep = 1, adapt = 2000, samples = 1000,
                thin = 2, span = 0.1)
plot_fit(hfit.s)

## End(Not run)
```

`simulate` *Simulate from the DCRW model with Argos location errors*

Description

For testing bsam models

Usage

```
simulate(Nt = 100, gamma = 0.8, Sigma = matrix(c(0.01, 0, 0, 0.01), 2, 2),
  amf = tpar())
```

Arguments

`Nt`	number of time steps to simulate
`gamma`	move persistence parameter
`Sigma`	variance-covariance matrix for movement process
`amf`	Argos error data, defined by default via the `tpar` function which uses the t-distribution scale and df estimates from Jonsen et al (2005)

Value

a data_frame of true locations and locations with Argos error

ssm — *Fits state-space models to Argos data*

Description

Takes output from `dat4jags`, sets up initial values, calls JAGS, and aggregates results. Intended for internal use, called by `fit_ssm`.

Usage

```
ssm(d, model = "DCRW", adapt, samples, thin, chains, span)
```

Arguments

`d`	structured data from `dat4jags` to be passed to JAGS
`model`	the state-space model to be fit: DCRW or DCRWS
`adapt`	number of samples in adaptation/burnin phase
`samples`	number of posterior samples
`thin`	thinning factor to reduce posterior sample autocorrelation
`chains`	number of parallel McMC chains to run
`span`	span

Value

Returns a list of McMC samples from marginal posteriors and a summary `data.frame` of mean and median position estimates.

See Also

Function to be called by `fit_ssm`.

tpar — *ARGOS Error Fixed Parameters*

Description

ARGOS Error Fixed Parameters for Location Classes

Usage

```
tpar()
```

Details

This is an internal function used by `dat4jags` to specify measurement error parameters.

These are the fixed parameters (t-distribution scale & df) for ARGOS error classes, from Jonsen et al (2005) Ecology 86:2874-2880.

Value

A dataframe with columns

`lc`	ARGOS location class as an ordered factor
`itau2.lon`	precision parameters for longitude in degrees
`itau2.lat`	precision parameters for latitude in degrees
`nu.lon`	df parameters for longitude
`nu.lat`	df parameters for latitude

Index