Package: movetrack (via r-universe)

September 3, 2024

Title Estimate flight tracks from telemetry data

Description `movetrack` is a `R` package that provides simple functionality to estimate flight tracks from telemetry data using random walk models written in Stan.

Version 0.3.2

License MIT + file LICENSE

URL https://g-rppl.github.io/movetrack/, https://github.com/g-rppl/movetrack

BugReports https://github.com/g-rppl/movetrack/issues

Depends R (>= 4.0.0)

Imports HDInterval, cmdstanr, dplyr, lubridate, ggplot2

Remotes stan-dev/cmdstanr

SystemRequirements CmdStan (https://mc-stan.org/users/interfaces/cmdstan)

Encoding UTF-8

VignetteBuilder knitr

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Suggests knitr, rmarkdown, sf, leaflet, sfheaders, DiagrammeR, testthat (>= 3.0.0)

Config/testthat/edition 3

LazyData true

Repository https://g-rppl.r-universe.dev

RemoteUrl https://github.com/g-rppl/movetrack

RemoteRef HEAD

RemoteSha 8ac2a4af52df7a0a729885d77a673824406e0502

Contents

`as.data.frame.movetrack` *Coerce to a Data Frame*

Description

Coerce a `movetrack` object to a `data.frame`.

Usage

```
## S3 method for class 'movetrack'
as.data.frame(x, ...)
```

Arguments

`x`	An object of class `movetrack`.
`...`	Unused; for compatibility with the generic method.

See Also

`summary.movetrack()`

`getDraws` *Extract draws*

Description

Extract draws from a `movetrack` object.

Usage

```
getDraws(fit, nsim = 50)
```

Arguments

fit	An object of class movetrack.
nsim	The number of simulations to extract; defaults to 50.

Value

A data.frame with the draws.

---

locate *Estimate locations*

Description

Calculate point estimates based on antenna bearing and signal strength.

Usage

```
locate(
  data,
  ID = "tagDeployID",
  ts = "ts",
  sig = "sig",
  aLon = "recvDeployLon",
  aLat = "recvDeployLat",
  aType = NULL,
  aBearing = "antBearing",
  aRange = 12,
  dTime = 2
)
```

Arguments

data	A data.frame containing the telemetry data.
ID	Unique identifier for individuals or tag deployments.
ts	Timestamp column.
sig	Signal strength column.
aLon	Antenna longitude column.
aLat	Antenna latitude column.
aType	Antenna type column, only required for antenna-specific detection ranges.
aBearing	Antenna bearing column.
aRange	Assumed maximum detection range of antennas in kilometres. Can be a single value or a named list of values for different antenna types.
dTime	Time interval in minutes for which point estimates are to be calculated.

Details

This function performs the following steps as described in Baldwin et al. 2018:

- Estimate locations for each detection: half of the maximum detection range `aRange` along the directional beam.
- Derive oscillating measurement error arising from antenna geometry and orientation.
- Calculate weighted means (by signal strength) for each time interval `dTime`.

Value

Returns a `data.frame` containing estimated coordinates and measurement errors for each time interval together with the proportions of time intervals `w`.

Examples

```
## Not run:
data(motusData)
locate(motusData)
locate(motusData, dTime = 1, aRange = 10)
locate(motusData, aType = "antType", aRange = list("yagi-5"=10, "yagi-6"=12))

## End(Not run)
```

`mapTrack` *Map model result*

Description

Map individual flight trajectories and model uncertainty.

Usage

```
mapTrack(fit, id = NULL, nsim = 50, lwd = 2, alpha = 0.1)
```

Arguments

`fit`	An object of class `movetrack`.
`id`	The individuals to plot. Defaults to `NULL` which plots all individuals.
`nsim`	The number of posterior draws to plot. Defaults to `50`.
`lwd`	The line width for the mean trajectory. Defaults to `2`.
`alpha`	The alpha value for the posterior draws. Defaults to `0.1`.

Value

Returns an overview map with the mean trajectories and `nsim` posterior draws per individual.

Examples

```
## Not run:
mapTrack(fit)
mapTrack(fit, nsim = 100, alpha = 0.05)

## End(Not run)
```

`motusData` *Motus test data*

Description

Motus test data

Usage

```
motusData
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 4464 rows and 7 columns.

References

https://motus.org/

`plot.movetrack` *Plot model results*

Description

Plot model results per individual and variable.

Usage

```
## S3 method for class 'movetrack'
plot(x, vars = c("lon", "lat"), id = NULL, ...)
```

Arguments

`x`	An object of class `movetrack`.
`vars`	The variables to plot. Defaults to `c("lon", "lat")`.
`id`	The individuals to plot. Defaults to `NULL` which plots all individuals.
`...`	Additional arguments passed to `movetrack::summary()`.

Value

Returns one or multiple ggplot plots.

See Also

movetrack::summary()

Examples

```
## Not run:
# Set ggplot theme
theme_set(theme_bw(base_size = 20))

# Plot
plot(fit)
plot(fit, vars = "speed", prob = 0.89, ci = "ETI")

## End(Not run)
```

---

print.movetrack *Print*

Description

Print a summary for a movetrack object.

Usage

```
## S3 method for class 'movetrack'
print(x, digits = 3, ...)
```

Arguments

x	An object of class movetrack.
digits	The minimal number of *significant* digits; defaults to 3.
...	Additional arguments passed to print().

See Also

summary.movetrack()

summary.movetrack *Summary*

Description

Create a summary of a movetrack object.

Usage

```
## S3 method for class 'movetrack'
summary(object, var = c("lon", "lat"), ci = "HDI", prob = 0.9, ...)
```

Arguments

object	An object of class movetrack.
var	The variable to summarise; defaults to 'lon'.
ci	The method used to calculate the credible intervals. Available options are 'HDI' for the highest posterior density interval and 'ETI' for the equal-tailed interval; defaults to 'HDI'.
prob	The probability mass of the credible interval; defaults to 0.9.
...	Unused; for compatibility with the generic method.

Value

A data.frame with the summary statistics.

Examples

```
## Not run:
summary(fit)
summary(fit, var = "distance")
summary(fit, ci = "ETI", prob = 0.89)

## End(Not run)
```

track *Model data*

Description

Model flight path from point estimates using a Hidden Markov Model.

Usage

```
track(data, states = 1, i_lambda = TRUE, ...)
```

Arguments

data	A data.frame containing the point estimate data.
states	The number of states to use in the model; defaults to 1.
i_lambda	Logical indicating whether to estimate individual correlation parameters; defaults to TRUE.
...	Additional arguments passed to cmdstanr::sample().

Details

This function calls Stan via cmdstanr and uses a Hidden Markov Model (HMM) to estimate individual flight paths.

Value

Returns a movetrack object including the posterior distributions for longitude, latitude, distance, and speed per time interval.

References

Auger-Méthé, M., Newman, K., Cole, D., Empacher, F., Gryba, R., King, A. A., ... & Thomas, L. (2021). A guide to state–space modeling of ecological time series. *Ecological Monographs*, 91(4), e01470. doi:10.1002/ecm.1470

Baldwin, J. W., Leap, K., Finn, J. T., & Smetzer, J. R. (2018). Bayesian state-space models reveal unobserved off-shore nocturnal migration from Motus data. *Ecological Modelling*, 386, 38-46. doi:10.1016/j.ecolmodel.2018.08.006

Jonsen, I. D., Flemming, J. M., & Myers, R. A. (2005). Robust state–space modeling of animal movement data. *Ecology*, 86(11), 2874-2880. doi:10.1890/04-1852

See Also

cmdstanr::sample()

Examples

```
## Not run:
# Load data
data(motusData)

# Estimate locations
loc <- locate(motusData, dTime = 2)

# Model flight paths
track(loc, states = 2, parallel_chains = 4)
track(loc, i_lambda = FALSE, parallel_chains = 4)

## End(Not run)
```

Index