Package: ctmmUtils (via r-universe)

September 8, 2024

Type Package

Title Auxillary functions for using the {ctmm} package efficiently

Version 0.0.0.9007

Date 2024-04-11

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

Description Utility functions to augment the the {ctmm} package. The additional function are targeted toward analysis of marine mammal telemetry, but can be used or easily modified for other situations.

License CC0

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

Imports dplyr, lubridate, ctmm, janitor, tibble, tidyr, trip, readr, methods, mgcv, sf, units, utils, plotly, ggplot2, fuzzyjoin, progressr, doFuture, foreach

Suggests pathroutr

Remotes jmlondon/pathroutr

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

RemoteUrl https://github.com/dsjohnson/ctmmUtils

RemoteRef HEAD

RemoteSha 39c503155b15574f2b7d563da4b6943a5c604aa1

Contents

`crawlUtils-package` *Functions To Increase Usability Of The* `ctmm` *Package*

Description

This package is a collection of functions that enhance the `ctmm` package for for analysis of animal telemetry data.

Package:	crawlUtils
Type:	Package
Version:	0.0.0.9007
Date:	April 11, 2024
License:	CC0
LazyLoad:	yes

Note

This software package is developed and maintained by scientists at the NOAA Fisheries Pacific Islands Fisheries Science Center and should be considered a fundamental research communication. The recommendations and conclusions presented here are those of the authors and this software should not be construed as official communication by NMFS, NOAA, or the U.S. Dept. of Commerce. In addition, reference to trade names does not imply endorsement by the National Marine Fisheries Service, NOAA. While the best efforts have been made to insure the highest quality, tools such as this are under constant development and are subject to change.

Author(s)

Devin S. Johnson and Josh M. London Maintainer: Devin S. Johnson <devin.johnson@noaa.gov>

`as_telem` *Convert Wildlife Computers data imported with ‘read_wc_dirs‘ to a ‘telemetry‘ object from the ‘ctmm‘ package.*

Description

Convert Wildlife Computers data imported with ‘read_wc_dirs‘ to a ‘telemetry‘ object from the ‘ctmm‘ package.

Usage

```
as_telem(x, ...)
```

Arguments

`x`	An sf data frame output by the function ‘read_wc_dirs‘.
`...`	Additional arguments to be passed to ‘ctmmas.temeletry‘

Author(s)

Josh M. London, Devin S. Johnson

`ctmm_select_parallel` *Parallel fitting of ctmm OUF family models*

Description

Parallel fitting of ctmm OUF family models

Usage

```
ctmm_select_parallel(tdata, add_ess = FALSE, ...)
```

Arguments

`tdata`	A list of telemetry data objects.
`add_ess`	Calculated "mutual information" and "regression" effective sample sizes and weights from Bartoszek (2016).
`...`	Additional arguments (besides ‘data‘ and ‘CTMM‘!) passed to ‘ctmm.select‘

References

Bartoszek, K. (2016). Phylogenetic effective sample size. Journal of Theoretical Biology, 407, 371-386. (See https://arxiv.org/pdf/1507.07113.pdf).

`migration_det` *Migration detection*

Description

Creates a data table that indicates the times of different phases of movement. This method uses changes in the overall dispersion rate of the animal from the 'base' time to detect changes in overall movement from small scale local movement to large scale migration.

Usage

```
migration_det(
  data,
  min_disp,
  max_num_mig = 1,
  min_phase_len = 3,
  grid_res = "day",
  base = "first",
  max_k = 100
)
```

Arguments

`data`	A 'telemytry' data object of locations (see 'as.telemetry')
`min_disp`	The minimum dispersion rate to be considered a migration interval, e.g. 10 for a 10km dispersion minimum.
`max_num_mig`	The maximum number of migration intervals.
`min_phase_len`	The minimum length of time that a migration or non-migration event will take, e.g., 7 implies a minimum of 7 time intervals for a phase.
`grid_res`	The temporal resolution at which migrations are detected. e.g., "day" (default) implies migration start and end is detected on a daily resolution.
`base`	The location at which dispersion is measured. Can be one of `"first"` (first location), `"last"` (final location), or some other `sf::sfc` point location.
`max_k`	The maximum degrees of freedom used by `mgcv::gam` to model dispersion and estimate the derivative of the dispersion function.

Author(s)

Devin S. Johnson

`ouf_corfun` *Create covariance function for a fitted OU or OUF ctmm model object*

Description

A function is created to evaluate the covariance function of the fitted OUF movement model

Usage

```
ouf_corfun(x)
```

Arguments

`x`	A ‘ctmm‘ object created by a call to `ctmm.fit` or `ctmm.select`

Details

The function returns a function to evaluate the covariance of the fitted Ornstein-Ulenbeck Foraging movement model. The returned function has 2 arguments: (1) `s` and (2) `t` both vectors of times to evaluate the covariance function of the fitted OUF model.

Author(s)

Devin S. Johnson

`ouf_covmat` *Calculate correlation matrix for a set of times from a ‘ctmm‘ covariance function*

Description

Using a correlation function created by `ouf_corfun` from a fitted OUF related model a covariance (correlation) matrix is created for observations at the user provided times.

Usage

```
ouf_covmat(x, times, inverse = FALSE)
```

Arguments

`x`	A `ctmm` model object from a call to `ctmm.fit` or ‘ctmm.select‘.
`times`	A vector of POSIX times at which the covariance matrix will be constructed. A ‘telemetry‘ data object will also work.
`inverse`	Logical. Should the inverse covariance matrix be returned. Defaults to ‘FALSE‘

Author(s)

Devin S. Johnson

`ouf_ess` *Calculate Effective Sample Size for a Set of OUF locations*

Description

Estimates the number of independent locations in a ‘ctmm‘ data set using the mutual information method of Bartoszek (2016).

Usage

```
ouf_ess(x, times)
```

Arguments

`x`	A `ctmm` object (See `ctmm.fit`).
`times`	A vector of POSIX times. A ‘telemtry‘ data object will also work.

Details

This function uses the "mutual information" effective sample size of Bartoszek (2016) to calculate the equivalent number of independent animal locations. It also calculates individual contributions of each location using the regression effective sample size in Bartoszek (2016). The output is a named list with ‘Ne‘ equal to the overall sample and ‘w‘ is a vector of weights that sum to 1 overall. If you want the ESS value of each observation ‘Ne * w‘ will provide it.

Author(s)

Devin S. Johnson

References

Bartoszek, K. (2016). Phylogenetic effective sample size. Journal of Theoretical Biology. 407:371-386. (See https://arxiv.org/pdf/1507.07113.pdf).

`ouf_ess_parallel` *Parallel ESS calculation of ctmm OUF family models*

Description

Parallel ESS calculation of ctmm OUF family models

Usage

```
ouf_ess_parallel(fits, tdata, ...)
```

Arguments

`fits`	A list of fitted ‘ctmm‘ OUF family models
`tdata`	A list of telemetry data objects.
`...`	Additional arguments (besides ‘data‘ and ‘CTMM‘!) passed to ‘ctmm.select‘

`plot_disp` *Plot migration detection results*

Description

PLot the animal dispersion from the base location over time. Points are colored to reflect estimated migration and non-migration phases. See `migration_det`

Usage

```
plot_disp(data, migr_tbl, interactive = FALSE)
```

Arguments

`data`	Original data used by ‘migration_det‘ call.
`migr_tbl`	Results table produced by ‘migration_det‘.
`interactive`	Logical. If ‘TRUE‘ the ‘plotly‘ package will be used to make the plot more interactive.

Author(s)

Devin S. Johnson

`read_wc_dirs` *Read individual telemetry data from Wildlife Computers portal directories*

Description

Read and combine data downloaded from Wildlife Computers portal into individual directories.

Usage

```
read_wc_dirs(x, remove_duplicates = TRUE)
```

Arguments

`x`	Directory containing the individual telemetry data directories.
`remove_duplicates`	Logical. Should observations with duplicated times be removed? The observation with the highest quality will be retained.

Author(s)

Devin S. Johnson, Josh M. London

`telem_sda` *Filter track for speed, distance and angle.*

Description

Applies the Freitas SDA (speed, distance, angle) filter to assess telemetry data outliers. The ‘sda‘ function from the ‘trip‘ package is used for computation.

Usage

```
telem_sda(data, ..., filter = TRUE)
```

Arguments

`data`	‘telemetry‘ data object from the ‘ctmm‘ package.
`...`	Additional arguments for the ‘link[trip]sda‘ function. The argument ‘smax‘ (in km/hr) is necessary. Please see the ‘link[trip]sda‘ documentation.
`filter`	Logical. Remove locations determined to be outliers based on the SDA filter. If ‘filter=FALSE‘ then a column named ‘keep‘ will indicate which observations should be kept, but all will remain in the data. The default is ‘filter=TRUE‘

Author(s)

Devin S. Johnson

References

Freitas, C., Lydersen, C., Fedak, M. A. and Kovacs, K. M. (2008), A simple new algorithm to filter marine mammal Argos locations. Marine Mammal Science, 24: 315?V325. doi: 10.1111/j.1748-7692.2007.00180.x

Index