

Package: ctmUtils (via r-universe)

September 30, 2026

Type Package

Title Auxillary functions for using the {ctmm} package efficiently

Version 0.0.0.9111

Date February, 25, 2026

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

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

Imports dplyr, lubridate, ctm, janitor, readr, methods, terra, mgcv, sf, units, utils, plotly, ggplot2, fuzzyjoin, progressr, doFuture, foreach

Suggests pathroutr

Remotes jmlondon/pathroutr

Config/pak/sysreqs

libabsl-dev cmake libgdal-dev gdal-bin libgeos-dev libgmp3-dev make libgsl0-dev libicu-dev libuv1-dev libmpfr-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev libx11-dev

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

Date/Publication 2026-07-02 19:47:39 UTC

RemoteUrl <https://github.com/dsjohnson/ctmUtils>

RemoteRef HEAD

RemoteSha 9879a4144eaa09b4efe7746b619a6aae85586a5f

Contents

ctmmUtils-package	2
akde_barrier	3
as_telem	4
ctmm_select_parallel	4
hex_size	5
migration_det	5
ouf_corfun	6
ouf_covmat	7
ouf_ess	7
ouf_ess_parallel	8
plot_disp	8
read_wc_dirs	9
set_default_ellipses	10
st_bbox_list	10
st_not_within	11
st_to_360	11
st_union_list	12
ud_area_table	12
Index	13

ctmmUtils-package	<i>Functions To Increase Usability Of The ctmm Package</i>
-------------------	--

Description

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

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>

akde_barrier

*Safe method for calculating an AKDE with a barrier***Description**

Safe method for calculating an AKDE with a barrier

Usage

```
akde_barrier(
  data,
  CTMM,
  VMM = NULL,
  R = list(),
  barrier = NULL,
  barrier_in = FALSE,
  debias = TRUE,
  weights = FALSE,
  smooth = TRUE,
  error = 0.001,
  res = 10,
  grid = NULL,
  ...
)
```

Arguments

data	2D timeseries telemetry data represented as a telemetry object or list of objects.
CTMM	A ctmm movement model from the output of ctmm.fit or list of objects.
VMM	An optional vertical ctmm object for 3D home-range calculation.
R	A named list of raster covariates if CTMM contains an RSF model.
barrier	An 'sf' polygon data frame for enforcing hard boundaries
barrier_in	Locations are assumed to be inside the 'sf' polygons if 'barrier_in=TRUE' and outside of 'barrier' if 'barrier_in=FALSE'.
debias	Debias the distribution for area estimation (AKDEc).
weights	Optimally weight the data to account for sampling bias (See ' bandwidth ' for akde details).
smooth	"Smooth" out errors from the data.
error	Target probability error.
res	Number of grid points along each axis, relative to the bandwidth.
grid	Optional grid specification via raster, UD, or list of arguments (See 'Details' in ' akde ').
...	Arguments passed to akde, ' bandwidth ', and ' mean.ctmm '.

as_telem	<i>Convert Wildlife Computers data imported with ‘read_wc_dirs’ to a ‘telemetry’ object from the ‘ctmm’ package.</i>
----------	--

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 ‘ as.telemetry ’

Author(s)

Josh M. London, Devin S. Johnson

ctmm_select_parallel	<i>Parallel fitting of ctmm OUF family models</i>
----------------------	---

Description

Parallel fitting of ctmm OUF family models

Usage

```
ctmm_select_parallel(tdata, ...)
```

Arguments

tdata	A list of telemetry data objects.
...	Additional arguments (besides ‘data’ and ‘CTMM’!) passed to ‘ ctmm.select ’

hex_size	<i>Calculate cellsize value for hexagon grid</i>
----------	--

Description

Calculates the appropriate cellsize argument for making a hexagon grid with [st_make_grid](#).

Usage

```
hex_size(area = NULL, radius = NULL, sep = NULL)
```

Arguments

area	A value (m ²) for the resulting area of a full hexagon cell
radius	The value for the distance (m) from the centroids to the edge of full hexagon cells.
sep	The distance (m) between centroids of the hexagon grid.

Author(s)

Devin S. Johnson

References

See <https://github.com/r-spatial/sf/issues/1505>

migration_det	<i>Migration detection</i>
---------------	----------------------------

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 ‘telemetry’ 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 <code>sf::sfc</code> point location.
max_k	The maximum degrees of freedom used by <code>mgcv::gam</code> to model dispersion and estimate the derivative of the dispersion function.

Author(s)

Devin S. Johnson

ouf_corfun	<i>Create covariance function for a fitted OU or OUF ctmm model object</i>
------------	--

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-Uhlenbeck 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	<i>Calculate correlation matrix for a set of times from a 'ctmm' covariance function</i>
------------	--

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	<i>Calculate Effective Sample Size for a Set of OUF locations</i>
---------	---

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 'telemetry' 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	<i>Parallel ESS calculation of ctmm OUF family models</i>
------------------	---

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	<i>Plot migration detection results</i>
-----------	---

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 <code>'migration_det'</code> call.
migr_tbl	Results table produced by <code>'migration_det'</code> .
interactive	Logical. If <code>'TRUE'</code> the <code>'plotly'</code> package will be used to make the plot more interactive.

Author(s)

Devin S. Johnson

read_wc_dirs	<i>Read individual telemetry data from Wildlife Computers portal directories</i>
--------------	--

Description

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

Usage

```
read_wc_dirs(
  x,
  remove_duplicates = TRUE,
  default_ellipse = TRUE,
  rm_mote = 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.
default_ellipse	Logical. Should a default set of Argos error ellipse values be added for <code>'Fast-GPS'</code> and <code>'Argos'</code> doppler locations? See <code>'set_default_ellipses'</code> for further information.
rm_mote	Logical, default = <code>'TRUE'</code> . If set to <code>'rm_mote = TRUE'</code> the locations that are <code>'type == "Mote"'</code> stations will be removed, due to lack of location error information. To force retention, set <code>'rm_mote = FALSE'</code> .

Author(s)

Devin S. Johnson, Josh M. London

set_default_ellipses	<i>Set Default Values for Missing ARGOS Diagnostic Data</i>
----------------------	---

Description

This function will fill in missing ARGOS diagnostic ellipse data for either type of location, ‘Argos’ or ‘FastGPS’. This is necessary for converting the data to ‘telemetry’ format for the ‘ctmm’ package. The default values used for non-KF Argos data (old location classes) are those in Vincent et al. (2002; Table 1 nonfiltered). For ‘FastGPS’ data, the values in Dujon et al. (2014; Table 1, 95%-tile unfiltered.).

Because Dujon et al. (2014) only used error magnitude, we assumed normally distributed error positions, $N(0, \sigma^2)$. Thus, the error magnitudes are Rayleigh distributed. So we used the CDF of the Rayleigh distribution and the 95%-tile given in the table to solve for the appropriate σ .

Usage

```
set_default_ellipses(x)
```

Arguments

x	An sf data frame output by the function ‘ read_wc_dirs ’.
---	---

st_bbox_list	<i>sf::st_bbox for a list of sf or sfc objects.</i>
--------------	---

Description

sf::st_bbox for a list of sf or sfc objects.

Usage

```
st_bbox_list(x, union = TRUE, as_sfc = FALSE)
```

Arguments

x	A list of sf or sfc objects.
union	Logical. Should the bounding box of the union be returned instead of a list of bounding boxes.
as_sfc	Logical. Should the bounding box (boxes) be returned as sfc objects.

Author(s)

Devin S. Johnson

st_not_within	<i>Predicate function for st_filter</i>
---------------	---

Description

Predicate function to use with `st_filter` such that elements of one spatial object are selected if they are not contained at all in the other. See [sf::st_within](#)

Usage

```
st_not_within(x, y, sparse = TRUE, prepared = TRUE, ...)
```

Arguments

x	object of class sf, sfc or sfg
y	object of class sf, sfc or sfg; if missing, x is used
sparse	logical; should a sparse index list be returned (TRUE) or a dense logical matrix? See sf::st_within .
prepared	logical; prepare geometry for x, before looping over y? See sf::st_within .
...	passed on to <code>s2_options</code>

st_to_360	<i>Convert longitude/latitude coordinates from -180/180 to 0/360</i>
-----------	--

Description

Converts sf data with EPSG = 4326 from -180/180 specification to 0/360 for plotting with the mapview package etc.

Usage

```
st_to_360(x)
```

Arguments

x	An sf data frame with EPSG=4326.
---	----------------------------------

Author(s)

Josh London

st_union_list	sf::st_union for a list of sf or sfc objects.
---------------	---

Description

sf::st_union for a list of sf or sfc objects.

Usage

st_union_list(x)

Arguments

x A list of sf or sfc objects.

Author(s)

Devin S. Johnson

ud_area_table	Get UD Area Table
---------------	-------------------

Description

Get UD Area Table

Usage

ud_area_table(uds, level.UD = 0.95, unit = NULL)

Arguments

uds A named list of ‘ctmm’ ‘UD’ objects
level.UD A vector of UD levels for computation
unit Units for area estimates. Defaults to ‘unit = "m^2"’

Index

akde, [3](#)
akde_barrier, [3](#)
as.telemetry, [4](#), [6](#)
as_telem, [4](#)

bandwidth, [3](#)

ctmm.fit, [6](#), [7](#)
ctmm.select, [4](#), [6–8](#)
ctmm_select_parallel, [4](#)
ctmmUtils (ctmmUtils-package), [2](#)
ctmmUtils-package, [2](#)

hex_size, [5](#)

mean.ctmm, [3](#)
migration_det, [5](#), [8](#), [9](#)

ouf_corfun, [6](#), [7](#)
ouf_covmat, [7](#)
ouf_ess, [7](#)
ouf_ess_parallel, [8](#)

plot_disp, [8](#)

read_wc_dirs, [4](#), [9](#), [10](#)

set_default_ellipses, [9](#), [10](#)
sf::st_within, [11](#)
st_bbox_list, [10](#)
st_make_grid, [5](#)
st_not_within, [11](#)
st_to_360, [11](#)
st_union_list, [12](#)

ud_area_table, [12](#)