Package: animovement (via r-universe)

September 15, 2024

Type Package

Title An R toolbox for analysing animal movement across space and time

Version 0.2.0.9000

Description An R toolbox for analysing animal movement across space and time.

License MIT + file LICENSE

URL http://www.roald-arboel.com/animovement/,

https://github.com/roaldarbol/animovement/

BugReports https://github.com/roaldarbol/animovement/issues

Encoding UTF-8

LazyData true

Imports circular, cli, collapse, dplyr, lifecycle, rlang, stringi, vroom, zoo

Suggests knitr, rmarkdown, testthat (>= 3.0.0), here, readxl, stringr, tinyplot, tinytable, ggplot2, patchwork, performance, see, markdown, tidyr

Config/testthat/edition 3

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

VignetteBuilder knitr

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

RemoteUrl https://github.com/roaldarbol/animovement

RemoteRef HEAD

RemoteSha 98caf832634971dc7b9211e47409f26aeb77d5ad

Contents

---

calculate_kinematics   *Calculate kinematics*

Description

[Experimental]

Calculate kinematics.

Usage

```
calculate_kinematics(data)
```

Arguments

data   Data frame

Value

An data frame data frame with kinematics calculated

---

calculate_statistics   *Calculate summary statistics*

Description

[Experimental]

Calculate summary statistics for tracks

Usage

```
calculate_statistics(
  data,
  measures = "median_mad",
  straightness = c("A", "B", "C", "D")
)
```

Arguments

data	A kinematics data frame
measures	Measures of central tendency and dispersion. Options are median_mad (default) and mean_sd. See description for more information.
straightness	Which method to calculate path straightness. Choose between "A" (default), "B", "C"... or a combination (e.g. "c("A","B")"). See description for details about the different calculations.

Value

An data frame data frame with kinematics calculated

---

clean_kinematics   *Clean kinematics*

Description

[Experimental]

Usage

```
clean_kinematics(data)
```

Arguments

data	tidy movement data frame with kinematics

Value

a clean kinematics data frame

---

read_animalta *Read AnimalTA data*

Description

[Experimental]

Usage

```
read_animalta(path)
```

Arguments

path   An AnimalTA data frame

Value

a movement dataframe

---

read_deeplabcut *Read DeepLabCut data*

Description

[Experimental]

Usage

```
read_deeplabcut(data)
```

Arguments

data   A DeepLabCut data frame

Value

a movement dataframe

---

read_idtracker *Read idtracker.ai data*

Description

[Experimental]

Usage

```
read_idtracker(data)
```

Arguments

data   An idtracker.ai data frame

Value

a movement dataframe

---

read_movement *Read movement data*

Description

[Experimental]

Usage

```
read_movement(data)
```

Arguments

data   A movement data frame

Value

a movement dataframe

---

read_sleap   *Read SLEAP data*

Description

[Experimental]

Usage

```
read_sleap(data)
```

Arguments

data   A SLEAP data frame

Value

a movement dataframe

---

read_trackball   *Read trackball data*

Description

[Experimental]

Read trackball data from a variety of setups and configurations.

Usage

```
read_trackball(
  paths,
  setup = c("of_free", "of_fixed", "fictrac"),
  sampling_rate,
  col_time = "time",
  col_dx = "x",
  col_dy = "y",
  ball_calibration = NULL,
  ball_diameter = NULL,
  distance_scale = NULL,
  distance_unit = NULL,
  verbose = FALSE
)
```

Arguments

paths	Two file paths, one for each sensor (although one is allowed for a fixed setup, of_fixed).
setup	Which type of experimental setup was used. Expects either of_free, of_fixed or fictrac (soon).
sampling_rate	Sampling rate tells the function how long time it should integrate over. A sampling rate of 60(Hz) will mean windows of 1/60 sec are used to integrate over.
col_time	Which column contains the information about time. Can be specified either by the column number (numeric) or the name of the column if it has one (character). Should either be a datetime (POSIXt) or seconds (numeric).
col_dx	Column name for x-axis values
col_dy	Column name for y-axis values
ball_calibration	
	When running an of_fixed experiment, you may (but it is not necessary) provide a calibration factor. This factor is the number recorded after a 360 degree spin. You can use the calibrate_trackball function to get this number. Alternatively, provide the ball_diameter and a distance_scale (e.g. mouse dpcm).
ball_diameter	When running a of_fixed experiment, the ball diameter is needed together with either ball_calibration or distance_scale.
distance_scale	If using computer mice, you might be getting unit-less data out. However, computer mice have a factor called "dots-per-cm", which you can use to convert your estimates into centimeters.
distance_unit	Which unit should be used. If distance_scale is also used, the unit will be for the scaled data. E.g. for trackball data with optical flow sensors, you can use the mouse dots-per-cm (dpcm) of 394 by setting distance_unit = "cm" and distance_scale = 394.
verbose	If FALSE (default), suppress most warning messages.

Value

a movement dataframe

---

read_treadmill	*Read treadmill data*

Description

[Experimental]

Usage

```
read_treadmill(data)
```

Arguments

data	A treadmill data frame

Value

a movement dataframe

---

read_trex	*Read TRex data*

Description

[Experimental]

Usage

```
read_trex(data)
```

Arguments

data	A TRex data frame

Value

a movement dataframe

---

smooth_track	*Smooth tracks*

Description

[Experimental]

Filtering/smoothing tracks is standard practice to root out noise in movement data. Here we provide some filter functions to do this. The function expects the data to be in the standard format, containing at least x, y and time variables.

Usage

```
smooth_track(data, method = c("rolling_median"), window_width = 5)
```

Arguments

data	Data frame
method	Which smoothing method to use. Currently only rolling mean is implemented ("rolling_mean").
window_width	How many observations to use for rolling window filters (e.g. "rolling_mean" or "rolling_median").

Value

A movement data frame

Index