Package 'zeitgebr'

May 14, 2018

Title Analysis of Circadian Behaviours

Date 2018-05-15

Version 0.3.1

Description Use behavioural variables to compute period, rhythmicity and other circadian parameters.

Depends R (>= 3.00),
  behavr

Imports data.table,
  lomb,
  ggplot2,
  pracma

Suggests testthat,
  covr,
  knitr

License GPL-3

Encoding UTF-8

LazyData true

URL https://github.com/rethomics/zeitgebr

BugReports https://github.com/rethomics/zeitgebr/issues

RoxygenNote 6.0.1

Roxygen list(markdown = TRUE)

R topics documented:

dams_sample	*A behavr table with approximately ten days of DAM2 recording for 32 fruit flies. The first 10, the following 11 and the last 11 animals have long, short and wild type period, respectively (see* `meta(dams_sample)`*).*

Description

A behavr table with approximately ten days of DAM2 recording for 32 fruit flies. The first 10, the following 11 and the last 11 animals have long, short and wild type period, respectively (see `meta(dams_sample)`).

Usage

```
dams_sample
```

Format

An object of class behavr (inherits from `data.table`, `data.frame`) with 415040 rows and 3 columns.

Author(s)

Luis Garcia

References

Raw data stored at https://github.com/rethomics/zeitgebr/tree/master/raw_data

---

find_peaks	*Find peaks in a periodogram*

Description

This function locates the peaks in a pregenerated periodogram. Detection is based on [pracma::findpeaks](). Only the significant (i.e. `power > signif_threshold`) peaks are extracted.

Usage

```
find_peaks(data, n_peaks = 3)
```

Arguments

data	[behavr::behavr]() table representing a periodogram, as returned by [periodogram]()
n_peaks	maximal numbers of peak to be detected

Value

[behavr::behavr](#) table that is data with an extra column peak. peak is filled with zeros except for rows match a peak. In which case, rows have an integer value corresponding to the rank of the peak (e.g. 1 for the first peak).

References

- [zeitgebr tutorial](#) – the relevant rehtomics tutorial

See Also

- [periodogram](#) – to generate a periodogram in a first place

Examples

```
data(dams_sample)
per_dt_xs <- periodogram(activity, dams_sample, FUN = chi_sq_periodogram)
per_dt_xs_with_peaks <- find_peaks(per_dt_xs)
per_dt_xs_with_peaks[peak == 1]
```

---

periodogram *Computes periodograms*

Description

This function builds periodograms, with one of several methods, for each individual of a [behavr](#) table

Usage

```
periodogram(var, data, period_range = c(hours(16), hours(32)),
  resample_rate = 1/mins(15), alpha = 0.01, FUN = chi_sq_periodogram, ...)
```

Arguments

var	variable to analyse
data	[behavr](#) table
period_range	vector of size 2 defining minimal and maximal range of period to study (in seconds)
resample_rate	frequency to resample (up or down) the data at (in hertz)
alpha	significance level
FUN	function used to compute periodogram (see [periodogram_methods](#))
...	additional arguments to be passed to FUN

Value

A [behavr::behavr](#) table. In addition to the metadata, it contains data that encodes a periodogram (i.e. power vs period). The data contains the columns:

- power – the power the or equivalent (according to FUN)
- period – the period at which power is computed (in seconds)
- p_value – the p value associated to the power estimation
- signif threshold – the threshold above which power is considered significant

References

- [zeitgebr tutorial](#) – the relevant rehtomics tutorial

See Also

- [periodogram_methods](#) – the list of built-in methods
- [find_peaks](#) – to find peaks in the periodogram

Examples

```
data(dams_sample)
pdt <- periodogram(activity, dams_sample, FUN = ls_periodogram, oversampling = 4)
pdt <- periodogram(activity, dams_sample, FUN = chi_sq_periodogram)
```

---

periodogram_methods   *Methods For Computing Periodograms*

Description

These functions provides a series of methods to assess periodicity of circadian processes.

Usage

```
ac_periodogram(x, period_range = c(hours(16), hours(32)),
  sampling_rate = 1/mins(1), alpha = 0.05)

chi_sq_periodogram(x, period_range = c(hours(16), hours(32)),
  sampling_rate = 1/mins(1), alpha = 0.05, time_resolution = hours(0.1))

fourier_periodogram(x, period_range = c(hours(16), hours(32)),
  sampling_rate = 1/mins(1), alpha = 0.05)

ls_periodogram(x, period_range = c(hours(16), hours(32)),
  sampling_rate = 1/mins(1), alpha = 0.05, oversampling = 8)
```

Arguments

x	numeric vector
period_range	vector of size 2 defining minimal and maximal range of period to study (in seconds)
sampling_rate	the – implicitly regular – sampling rate of x (in hertz)
alpha	significance level
time_resolution	
	the resolution of periods to scan
oversampling	the oversampling factor (see lomb::lsp)

Value

a data.table with the columns:

- period – the period (in s)
- power – the power (or equivalent) for a given period
- p_value – the significance of the power
- signif_threshold – the significance threshold of the power (at alpha)

References

- zeitgebr tutorial – the relevant rehtomics tutorial

See Also

- lomb::lsp – the orginal function for ls_periodogram
- xsp::chiSqPeriodogram – code modified from
- stats::acf – the orginal function for ac_periodogram

Index