Package 'damr'

October 15, 2020

Title Interface to Drosophila Activity Monitor System Result Files

Date 2020-09-14

Version 0.3.7

Description Loads behavioural data from the widely used Drosophila Activity Monitor System (DAMS, TriKinetics <https://trikinetics.com/>) into the rethomics framework.

Depends R (>= 3.00),
behavr,
data.table

Imports readr

Suggests testthat,
covr,
knitr,
ggetho,
zeitgebr

License GPL-3

Encoding UTF-8

LazyData true

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

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

RoxygenNote 7.1.1

Roxygen list(markdown = TRUE)

R topics documented:

---

damr_example   *Get path to damr example*

Description

damr comes with a sample DAM2 files in its `inst/extdata` directory. `damr_example` allow make them easy to access.

Usage

```
damr_example(path = NULL)

damr_example_dir()
```

Arguments

path   Name of file. If `NULL`, the example files will be listed.

Author(s)

Hadley Wickham (modified from readr)

Examples

```
# list all files
damr_example()
# get path to one file
damr_example("M014.txt")
# get the directory wih all the files
damr_example_dir()
```

---

link_dam_metadata   *Link DAM2 or DAM5 metadata to result files*

Description

This function checks and add columns to DAMS metadata. This way, it can subsequently be loaded (via [load_dam](#)).

Usage

```
link_dam_metadata(x, result_dir)
```

Arguments

x   object such as a [data.frame](#), or the name of a file (see detail)

result_dir   the root directory where all daily data are saved

Details

These function will augment metadata from two different types of inputs:

1. A [data.frame](#) (recommended) In this case, the function will try to match requested data with data available on result_dir. The provided [data.frame](#) or [data.table](#) has typically one row per requested individual and the columns (not necessarily in this order):

   - file – the name of a data file (e.g. "Monitor3.txt"), it has to exists in result_dir.
   - start_datetime – the first day **and time** of the requested experiment (e.g. "2014-12-28 18:00:00").
   - stop_datetime – the last day and time of the requested experiment (e.g. "2014-12-30 19:00:00" or simply "2014-12-30").
   - region_id – the channel (between 1 and 32) in which the animal was in (e.g. 20). region_id is optional. If not provided, all 32 channels are loaded *with the same conditions*.
   - ??? *any number of arbitrary columns* to associate conditions/treatments/genotypes/... to the previous columns.

2. The name of a CSV file that contains a table as described in 1.

The time in data is expressed relatively to start_date. In other words, if you do circadian analysis, and your D -> L transitions are at 09:00:00, you want to set start_datetime = "YYY-MM-DD 09:00:00". The result_directory`` is the folder conraining all result (.txt) files (for instance, result_dir = "C:/where/I/Store/my/txt/files/"`)

Value

a [data.table::data.table](#) with the same rows as x, and extra columns used for further data loading

References

- [the rethomics workflow](#) – on the concept of "linking"
- [metadata tutorial](#) – how to work with metadata

See Also

- [load_dam](#) – to subsequently load the actual data

---

load_dam   *Load DAMS data from one or several continuous text files*

Description

Uses "linked metadata" to load data from either single beam (DAM2) or multibeam (DAM5) arrays.

Usage

```
load_dam(metadata, date_format = "%d %b %y", FUN = NULL, ...)
```

Arguments

metadata	[data.table::data.table](#) used to load data (see detail)
date_format	How dates are formatted in the DAM result files (see [read_dam_file](#))
FUN	function (optional) to transform the data from each animal immediately after is has been loaded.
...	extra arguments to be passed to FUN

Details

The linked metadata should be generated using [link_dam_metadata](#).

Value

A [behavr::behavr](#) table. In addition to the metadata, it contains the data, whith the columns:

- id – autogenerated unique identifier, one per animal
- t – time
- activity – number of beam crosses

References

- [damr tutorial](#) – how to use this function in practice

See Also

- [behavr::behavr](#) – the class of the resulting object
- [read_dam_file](#) – to load data from a single file (without metadata)

Examples

```
# This is where our toy data lives
root_dir <- damr_example_dir()

# Metadata already made for us.
# It defines condition and genotype of each animal
data(single_file_metadata)
print(single_file_metadata)
# Linking:
metadata <- link_dam_metadata(single_file_metadata, root_dir)

# We find and load the matching data
dt <- load_dam(metadata)
print(dt)
# An example of the use of FUN,
# we load only the first few reads by run `head()` on each animal,
# on the fly (no pun intended)
dt <- load_dam(metadata, FUN = head)
print(dt)
```

| read_dam_file | *Reads data from a single DAM2 single beam or a DAM5 multibeam file* |

Description

This function retrieves activity data in a DAMS text file. It allows selection of a date range and channels (i.e. regions).

Usage

```
read_dam_file(
  path,
  region_id = 1:32,
  start_datetime = -Inf,
  stop_datetime = +Inf,
  tz = "UTC",
  date_format = "%d %b %y"
)
```

Arguments

path	location of the file to read (character)
region_id	vector of unique regions to read
start_datetime, stop_datetime	
	the start and the end of an the experiment (see details)
tz	the timezone (see OlsonNames for a list)
date_format	the format of the dates in the DAM file (see details)

Details

start_datetime and stop_datetime are formatted as "YYYY-MM-DD HH:MM:SS". start_datetime is used as the reference time (ZT0). Therefore, if you are interested in circadian analysis and D -> L transitions are at 10:00:00, you probably want to set start_datetime = "YYYY-MM-DD 10:00:00".

According to the acquisition system, the date format can be inconsistently formatted between DAM Systems. Specify the format using strptime syntax. For instance:

- "%d %b %y" – the default, to parse "15 Nov 2019"
- "%d-%m-%y" – to parse "15-11-2019"
- "%Y-%m-%d" – the default to parse "2019-11-15"

Value

A behavr table. The metadata contains an autogenerated id per animal. The data has the columns:

- id – autogenerated unique identifier, one per animal
- t – time
- activity – number of beam crosses

See Also

- [load_dam](#) – to load data from many files and biological conditions using metadata (the recommended alternative)

Examples

```
path <- damr_example("M064.txt")
dt <- read_dam_file(path, region_id = c(1:3), start_datetime = "2017-06-30 15:00:00")
print(dt)
```

---

single_file_metadata	*A simple toy metadata defining the experimental conditions of 32 animals monitored at the same time in a single DAM2 monitor. Each animal has its own channel (region_id), as well as a condition and genotype. It serves as an example for [link_dam_metadata](#).*

Description

A simple toy metadata defining the experimental conditions of 32 animals monitored at the same time in a single DAM2 monitor. Each animal has its own channel (region_id), as well as a condition and genotype. It serves as an example for [link_dam_metadata](#).

Usage

```
single_file_metadata
```

Format

An object of class `data.frame` with 32 rows and 6 columns.

Author(s)

Quentin Geissmann

---

two_files_metadata	*A toy metadata defining the experimental conditions of 64 animals monitored at the same time in two separate DAM2 monitors. Each animal has its own channel (region_id), as well as a condition and genotype. It serves as an example for [link_dam_metadata](#).*

Description

A toy metadata defining the experimental conditions of 64 animals monitored at the same time in two separate DAM2 monitors. Each animal has its own channel (region_id), as well as a condition and genotype. It serves as an example for [link_dam_metadata](#).

Usage

```
two_files_metadata
```

Format

An object of class data.frame with 64 rows and 6 columns.

Author(s)

Quentin Geissmann

Index