

Package ‘damr’

September 4, 2017

Title Read Drosophila Activity Monitor Data

Date 2017-07-26

Version 0.0.0.9000

Description Retrieve behavioural data from the widely used Drosophila Activity Monitors.

Depends R (>= 2.10)

Imports behavr,
data.table,
dplyr,
readr,
hms,
tidyr,
magrittr

Suggests testthat

License GPL-3

Encoding UTF-8

LazyData true

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

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

RoxygenNote 6.0.1

Roxygen list(markdown = TRUE)

R topics documented:

| | |
|-----------------------------|---|
| damr_example | 2 |
| query_dam2 | 2 |
| read_dam2_file | 4 |
| single_file_query | 5 |
| two_files_query | 5 |

| | |
|--------------|----------|
| Index | 7 |
|--------------|----------|

| | |
|--------------|---------------------------------|
| damr_example | <i>Get path to damr example</i> |
|--------------|---------------------------------|

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 with all the files
damr_example_dir()
```

| | |
|------------|---|
| query_dam2 | <i>Retrieves DAM2 data from one or several continuous files</i> |
|------------|---|

Description

Uses a query mechanism to get data from a DAM2 array. This is useful when using the default behaviour of TriKinetics software where data is simply appended to a single long file per monitor.

Usage

```
query_dam2(result_dir, query, FUN = NULL, ...)
```

Arguments

| | |
|------------|--|
| result_dir | the root directory where all daily data are saved |
| query | data.frame representing a formatted query used to request data (see detail) |
| FUN | function (optional) to transform the data from each animal immediately after it has been loaded. |
| ... | extra arguments to be passed to FUN |

Details

query must be a [data.frame](#) (or [data.table](#)) object. Conceptually, each row of the query describes one animal with one set of conditions (when region_id is specified), or in each monitor (when it is not). It must have the following columns:

- file – the location of a data file (e.g. "Monitor3.txt").
- 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.

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 10:00:00, you want to set start_datetime = "YYY-MM-DD 10:00:00".

The root directory is the folder where your files live. For instance, result_dir = "C:/where/I/Store/my/txt/files/"

Value

A [behavr](#) table. The metadata contains all the query columns and 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

[read_dam2_file](#) to load data from a single file (without a query).

Examples

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

# A query already made for us.
# It defines condition and genotype of each animal
data(single_file_query)
print(single_file_query)
```

```
# we find and load the matching data
dt <- query_dam2(root_dir,single_file_query)
print(dt)

# genotype and condition to our metadata:
print(dt[meta=TRUE])

# Just the first few reads, we run `head()` on each animal
dt <- query_dam2(root_dir, single_file_query, FUN=head)
print(dt)
```

read_dam2_file

Reads data from a single DAM2 file

Description

This function retrieves activity data in a DAM2 file. It allows selection of a date range and channels (i.e. regions).

Usage

```
read_dam2_file(path, region_id = 1:32, start_datetime = -Inf,
  stop_datetime = +Inf, tz = "UTC")
```

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) |

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".

Value

A [behave](#) 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

[query_dam2](#) to load data from many files and biological conditions using a query system

Examples

```
path <- damr_example("M064.txt")
dt <- read_dam2_file(path, region_id=c(1:3), start_datetime="2017-06-30 15:00:00")
print(dt)
```

| | |
|-------------------|---|
| single_file_query | <i>A simple toy query 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 query_dam2.</i> |
|-------------------|---|

Description

A simple toy query 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 [query_dam2](#).

Usage

```
single_file_query
```

Format

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

Author(s)

Quentin Geissmann

| | |
|-----------------|---|
| two_files_query | <i>A toy query 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 query_dam2.</i> |
|-----------------|---|

Description

A toy query 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 [query_dam2](#).

Usage

`two_files_query`

Format

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

Author(s)

Quentin Geissmann

Index

*Topic **datasets**

single_file_query, [5](#)

two_files_query, [5](#)

behavr, [3](#), [4](#)

damr_example, [2](#)

damr_example_dir (damr_example), [2](#)

data.frame, [3](#)

data.table, [3](#)

OlsonNames, [4](#)

query_dam2, [2](#), [5](#)

read_dam2_file, [3](#), [4](#)

single_file_query, [5](#)

two_files_query, [5](#)