Package: arctools (via r-universe)

October 11, 2024

Type Package

Title Processing and Physical Activity Summaries of Minute Level Activity Data

Version 1.1.6

Description Provides functions to process minute level actigraphy-measured activity counts data and extract commonly used physical activity volume and fragmentation metrics.

License GPL-3

Encoding UTF-8

Depends R (>= 2.10)

RoxygenNote 7.2.1

LazyData true

Suggests testthat, data.table, dplyr, knitr, rmarkdown, ggplot2

Imports lubridate, runstats

VignetteBuilder knitr

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

RemoteUrl https://github.com/martakarass/arctools

RemoteRef HEAD

RemoteSha 1a0cd82f002e9fea841780b63fb669a03245a1f1

Contents

`arctools-package` *arctools: processing and physical activity summaries of minute-level activity data*

Description

arctools provides ‘activity_stats()‘ function to process minute level actigraphy-measured activity counts data and extract commonly used physical activity volume and fragmentation metrics.

Details

To learn more about arctools, start with the vignettes: ‘browseVignettes(package = "arctools")‘

Author(s)

Maintainer: Marta Karas <marta.karass@gmail.com> (ORCID)

Authors:

- Jennifer Schrack (ORCID)
- Jacek Urbanek (ORCID)

`activity_stats` *Compute physical activity summaries of minute level activity data*

Description

Process minute level actigraphy-measured activity counts data and extract commonly used physical activity volume and fragmentation metrics.

Usage

```
activity_stats(
  acc,
  acc_ts,
  impute_missing = TRUE,
  sedentary_thresh = 1853,
  nonwear_0s_minimum_window = 90,
  validday_nonwear_maximum_window = 144,
  subset_minutes = NULL,
  exclude_minutes = NULL,
  subset_weekdays = NULL,
  in_bed_time = NULL,
  out_bed_time = NULL,
  adjust_out_colnames = TRUE
)
```

Arguments

acc	A numeric vector. A minute-level activity counts data vector.
acc_ts	A POSIXct vector. A minute-level time of acc data collection. We strongly recommended to use lubridate::ymd_hms() function to create acc_ts (see Examples below).
impute_missing	A logical scalar. Whether or not to perform missing data imputation (see Details). Default is TRUE.
sedentary_thresh	A numeric scalar. If an activity count value falls below it then a corresponding minute is characterized as sedentary; otherwise, a corresponding minute is characterized as active. Default is 1853.
nonwear_0s_minimum_window	A numeric scalar. A minimum number of consecutive minutes with 0 activity count to be considered non-wear.
validday_nonwear_maximum_window	In integer scalar. Maximum number of minutes of non-wear/not collecting data so as the day is still considered valid. Default is 144 (10% of 1440 minutes of a full day).
subset_minutes	Integer vector. Contains subset of a day's minutes within which activity summaries are to be computed. May take values from 1 (day's minute from 00:00 to 00:01) to 1440 (day's minute from 23:59 to 00:00). Default is NULL, i.e. no subset used (all day's minutes are used).
exclude_minutes	Integer vector. Contains subset of a day's minutes to be excluded from activity summaries computation. May take values from 1 (day's minute from 00:00 to 00:01) to 1440 (day's minute from 23:59 to 00:00). Default is NULL, i.e. no minutes excluded (all day's minutes are used).
subset_weekdays	Integer vector. Specfies days of a week within which activity summaries are to be computed. Takes values between 1 (Sunday) to 7 (Saturday). Default is NULL, i.e.no subset used (all days of a week are used).
in_bed_time	A POSIXct vector. An estimated in-bed time start. Together with a corresponding entry from out_bed_time vector, it defines a day-specific subset of "in bed time" minutes to be excluded from activity summaries computation. Default is NULL, i.e. no minutes excluded.
out_bed_time	A POSIXct vector. An estimated in-bed time end. Together with a corresponding entry from in_bed_time vector, it defines a day-specific subset of "in bed time" minutes to be excluded from activity summaries computation. Default is NULL, i.e. no minutes excluded.
adjust_out_colnames	A logical scalar. Whether or not to add an informative suffix to column names in the output data frame. This may happen in case any of the arguments: subset_minutes, or exclude_minutes, or in_bed_time and out_bed_time are set other than NULL. Default is TRUE.

Details

Physical activity statistics are aggregated from "valid" days, i.e. days with no more than 10 wear/non-wear detection algorithm closely following that of Choi et al. (2011). See `arctools::get_wear_flag()` for details.

Data imputation is recommended for valid days for non-wear time periods and is a default setting (see `impute_missing` arg). Count values are imputed from an "average day profile" – a minute-specific activity counts average computed across valid days within wear time.

Value

A data frame with physical activity summaries of minute level activity data. See README or vignette for summaries description.

References

Varma, V. R., Dey, D., Leroux, A., Di, J., Urbanek, J., Xiao, L., Zipunnikov, V. (2018). Total volume of physical activity: TAC, TLAC or TAC(lambda). Preventive medicine, 106, 233–235. https://doi.org/10.1016/j.ypmed.2017.10.028

Di, J., Leroux, A., Urbanek, J., Varadhan, R., Spira, A., Schrack, J., Zipunnikov, V. Patterns of sedentary and active time accumulation are associated with mortality in US adults: The NHANES study. https://doi.org/10.1101/182337

Choi, L., Liu, Z., Matthews, C. E., & Buchowski, M. S. (2011). Validation of accelerometer wear and nonwear time classification algorithm. Medicine and Science in Sports and Exercise. https://doi.org/10.1249/MSS.0b013e3181ed61a3

Koster, A., Shiroma, E. J., Caserotti, P., Matthews, C. E., Chen, K. Y., Glynn, N. W., & Harris, T. B. (2016). Comparison of Sedentary Estimates between activPAL and Hip- and Wrist-Worn ActiGraph. Medicine and science in sports and exercise, 48(8), 1514–1522. https://doi.org/10.1249/MSS.0000000000000924

Examples

```
fpath_i <- system.file("extdata", extdata_fnames[1], package = "arctools")
dat_i   <- as.data.frame(data.table::fread(fpath_i))
acc     <- dat_i$vectormagnitude
acc_ts  <- lubridate::ymd_hms(dat_i$timestamp)

## Example 1
## Summarize PA
activity_stats(acc, acc_ts)

## Example 2
## Summarize PA within minutes range corresponding to 12:00 AM - 6:00 AM
subset_12am_6am <- 1 : (6 * 1440/24)
activity_stats(acc, acc_ts, subset_minutes = subset_12am_6am)

## Example 3
## Summarize PA without (i.e., excluding) minutes range corresponding to 11:00 PM - 5:00 AM.
subset_11pm_5am <- c(
  (23 * 1440/24 + 1) : 1440,   ## 11:00 PM - midnight
  1 : (5 * 1440/24)            ## midnight - 5:00 AM
```

```
)
activity_stats(acc, acc_ts, exclude_minutes = subset_11pm_5am)
```

`extdata_fnames` *Names of exemplary accelerometry data file.*

Description

Names of exemplary accelerometry data files attached to the package. These data can be accessed via `system.file("extdata", "<data_file_name>.csv", package = "arctools")`.

Usage

```
extdata_fnames
```

Format

A character vector.

`get_actigraph_SN` *Get Actigraph device serial number*

Description

Read Actigraph device serial number from ActiLife accelerometry data file.

Usage

```
get_actigraph_SN(fpath_full)
```

Arguments

`fpath_full`	A string scalar. An absolute path to ActiLife accelerometry data file.

Value

String scalar. Actigraph device serial number.

Examples

```
fpath_full_i <- system.file("extdata", extdata_fnames[1], package = "arctools")
get_actigraph_SN(fpath_full_i)
```

`get_valid_day_flag` *Compute valid day flag*

Description

Compute valid/non-valid day flag (`1`/`0`) for each minute of activity counts data.

Usage

```
get_valid_day_flag(wear_flag, validday_nonwear_maximum_window = 144)
```

Arguments

`wear_flag`	An integer vector. The vector has value `1` if a minute belongs to a wear time-interval, value `0` if a minute belongs to a non-wear time-interval, and value `NA` to denote minutes before/after data collection started/finished. Vector `wear_flag` is assumed to be in midnight-to-midnight format, meaning its vector length is a multiple of number of minutes in a full day (1440). See `arctools::midnight_to_midnight()`, `arctools::get_wear_flag()`.
`validday_nonwear_maximum_window`	In integer scalar. Maxmimum number of minutes of non-wear/not collecting data so as the day is still considered valid. Default is `144` (10% of 1440 minutes of a full day).

Details

All minute-level observations from one day are assigned the same value of valid day flag. The flag is `1` if a day is determined to be valid, and `0` otherwise.

A day is determined to be valid if it has no more than `validday_nonwear_maximum_window` minutes of missing data. Data may be missing due to identified sensor nonwear or because activity data collection has not started yet/has finished already in a particular day.

Value

An integer vector. It has value `1` if a minute belongs to a valid day, and `0` otherwise.

Examples

```
## Read exemplary data
fpath_i <- system.file("extdata", extdata_fnames[1], package = "arctools")
dat_i   <- as.data.frame(data.table::fread(fpath_i))
acc     <- dat_i$vectormagnitude
acc_ts  <- lubridate::ymd_hms(dat_i$timestamp)
## Get acc data vector in "midnight_to_midnight" format
acc <- midnight_to_midnight(acc, acc_ts)
## Get wear/non-wear flag
wear_flag <- get_wear_flag(acc)
## Get valid/non-valid day flag
```

```
valid_day_flag <- get_valid_day_flag(wear_flag)
```

`get_wear_flag` *Compute wear/non-wear flag*

Description

Compute wear/non-wear flag (`1`/`0`) for each minute of activity counts data.

Usage

```
get_wear_flag(acc, nonwear_0s_minimum_window = 90)
```

Arguments

`acc`	A numeric vector. A minute-level activity counts data vector.
`nonwear_0s_minimum_window`	A numeric scalar. A minimum number of consecutive minutes with 0 activity count to be considered non-wear.

Details

Method implements wear/non-wear detection algorithm closely following that of Choi et al. (2011).

The wear/non-wear flag is determined based on activity counts data. A minute is classified as non-wear if it belongs to any `nonwear_0s_minimum_window` minutes-long interval of consecutive values 0 in activity counts data vector; here, "any interval" implies that a particular minute may be located at any location (beginning, middle, end) of interval of consecutive values 0 to be classified as a non-wear. Otherwise, a particular minute is classified as wear.

Similarly to recommendations in Discussion in Choi et al. (2011), the method assumes a threshold value of 0 for nonzero counts allowed during a nonwear time interval (I.e., no activity count equal >= 1 is allowed). The method also assumes 90 minutes as a default for minimum time of consecutive zero counts for a window to be flagged nonwear. Differently from recommendations in Discussion in Choi et al. (2011), it does not consider any "artifactual movement" interval of nonzero counts during a nonwear time interval.

Value

An integer vector. It has value `1` for a wear and `0` for non-wear flagged minute. It has the same vector length as `acc` vector. If there is an `NA` entry in `acc` vector, then the returned vector will have a corresponding entry set to `NA` too.

References

Choi, L., Liu, Z., Matthews, C. E., & Buchowski, M. S. (2011). Validation of accelerometer wear and nonwear time classification algorithm. Medicine and Science in Sports and Exercise. https://doi.org/10.1249/MSS.0b013e3181ed61a3

Examples

```
## Read exemplary data
fpath_i <- system.file("extdata", extdata_fnames[1], package = "arctools")
dat_i   <- as.data.frame(data.table::fread(fpath_i))
acc     <- dat_i$vectormagnitude
acc_ts  <- lubridate::ymd_hms(dat_i$timestamp)
## Get acc data vector in "midnight_to_midnight" format
acc <- midnight_to_midnight(acc, acc_ts)
## Get wear/non-wear flag
wear_flag <- get_wear_flag(acc)
```

`impute_missing_data` *Impute missing data*

Description

Impute missing data in minute-level activity counts data vector based on "average day profile".

Usage

```
impute_missing_data(
  acc,
  wear_flag,
  valid_day_flag,
  imputeFromValidDaysOnly = TRUE
)
```

Arguments

`acc`	A numeric vector. A minute-level activity counts data vector. It is assumed to be in midnight-to-midnight format, meaning its vector length is a multiple of number of minutes in a full day (1440; see `midnight_to_midnight()`).
`wear_flag`	An integer vector. Wear/non-wear flag (`1`/`0`) for each minute of activity counts data. It is assumed to be in midnight-to-midnight format, meaning its vector length is a multiple of number of minutes in a full day (1440). See `midnight_to_midnight()`, `get_wear_flag()`.
`valid_day_flag`	An integer vector. Valid/non-valid day flag (`1`/`0`) for each minute of activity counts data. It is assumed to be in midnight-to-midnight format, meaning its vector length is a multiple of number of minutes in a full day (1440). See `arctools::midnight_to_midnight()`, `arctools::get_valid_day_flag()`.
`imputeFromValidDaysOnly`	A logical scalar. Whether or not data from valid days only should be used for computing "average day profile" used for imputation.

Details

An "average day profile" is computed as average across minutes identified as wear and from valid days (see param. `imputeFromValidDaysOnly`). Activity counts data are imputed from "average day profile" for minutes identified as non-wear in days identified as valid, except for minutes before/after data collection start/end which remain `NA`.

Theoretically, it is possible that all valid days of data collection have non-wear flag for the some minute(s) (i.e., somebody is always taking off the watch for the same few minutes during a day) so there is no data to use to compute imputation values from. If it happens, then method uses 0 as imputation value(s).

Value

A numeric vector. A minute-level activity counts data vector with data imputed for minutes identified as non-wear in days identified as valid

Examples

```
## Read exemplary data
fpath_i <- system.file("extdata", extdata_fnames[1], package = "arctools")
dat_i   <- as.data.frame(data.table::fread(fpath_i))
acc     <- dat_i$vectormagnitude
acc_ts  <- lubridate::ymd_hms(dat_i$timestamp)
## Get acc data vector in "midnight_to_midnight" format
acc <- midnight_to_midnight(acc, acc_ts)
## Get wear/non-wear flag
wear_flag <- get_wear_flag(acc)
## Get valid/non-valid day flag
valid_day_flag <- get_valid_day_flag(wear_flag)
## Impute missing data in acc data vector
acc_imputed <- impute_missing_data(acc, wear_flag, valid_day_flag)
## Compare mean acc before/after imputation
c(mean(acc, na.rm = TRUE), mean(acc_imputed, na.rm = TRUE))
```

`midnight_to_midnight` *Expand activity data vector into midnight-to-midnight format*

Description

Expand activity data vector such that its length is a multiple of number of minutes in a full day (1440).

Usage

```
midnight_to_midnight(acc, acc_ts)
```

Arguments

acc	A numeric vector. A minute-level activity counts data vector.
acc_ts	A POSIXct vector. Time of activity data collection, corresponding to acc. We strongly recommended to use lubridate::ymd_hms() function to create acc_ts (see Examples below).

Details

In the returned vector, 1st observation corresponds to minute of 00:00-00:01 on the first day of data collection, and last observation corresponds to minute of 23:59-00:00 on the last day of data collection. Entries corresponding to no data in original activity data vector are filled with NA.

Value

A numeric vector. A minute-level activity counts data vector in midnight-to-midnight format.

Examples

```
## Read exemplary data
fpath_i <- system.file("extdata", extdata_fnames[1], package = "arctools")
dat_i   <- as.data.frame(data.table::fread(fpath_i))
acc     <- dat_i$vectormagnitude
acc_ts  <- lubridate::ymd_hms(dat_i$timestamp)
## Get acc data vector in "midnight_to_midnight" format
acc <- midnight_to_midnight(acc, acc_ts)
## Observe we have an integer number of days
length(acc) / 1440
```

---

summarize_PA *Compute physical activity summaries of minute level activity data.*

Description

Compute physical activity summaries of minute level activity data.

Usage

```
summarize_PA(
  acc,
  acc_ts,
  wear_flag,
  valid_day_flag,
  sedentary_thresh = 1853,
  subset_minutes = NULL,
  exclude_minutes = NULL,
  subset_weekdays = NULL,
  in_bed_time = NULL,
```

```
  out_bed_time = NULL,
  adjust_out_colnames = TRUE
)
```

Arguments

acc	A numeric vector. A minute-level activity counts data vector. It is assumed to be in midnight-to-midnight format, meaning its vector length is a multiple of number of minutes in a full day (1440). See `arctools::midnight_to_midnight()`.
acc_ts	A POSIXct vector. Time of activity data collection, corresponding to `acc` in its original format (not: midnight-to-midnight). We strongly recommended to use `lubridate::ymd_hms()` function to create `acc_ts` (see Examples below).
wear_flag	An integer vector. It has value `1` if a minute belongs to a wear time-interval, value `0` if a minute belongs to a non-wear time-interval, and value `NA` to denote minutes before/after data collection started/finished. See `arctools::get_wear_flag()`.
valid_day_flag	An integer vector. It has value `1` if a minute belongs to a valid day, and `0` otherwise. See `arctools::get_valid_day_flag()`.
sedentary_thresh	A numeric scalar. If an activity count value falls below it then a corresponding minute is characterized as sedentary; otherwise, a corresponding minute is characterized as active. Default is `1853`.
subset_minutes	Integer vector. Contains subset of a day's minutes within which activity summaries are to be computed. May take values from `1` (day's minute from 00:00 to 00:01) to `1440` (day's minute from 23:59 to 00:00). Default is `NULL`, i.e. no subset used (all day's minutes are used).
exclude_minutes	Integer vector. Contains subset of a day's minutes to be excluded from activity summaries computation. May take values from `1` (day's minute from 00:00 to 00:01) to `1440` (day's minute from 23:59 to 00:00). Default is `NULL`, i.e. no minutes excluded (all day's minutes are used).
subset_weekdays	Integer vector. Specfies days of a week within which activity summaries are to be computed. Takes values between 1 (Sunday) to 7 (Saturday). Default is `NULL`, i.e.no subset used (all days of a week are used).
in_bed_time	A POSIXct vector. An estimated in-bed time start. Together with a corresponding entry from `out_bed_time` vector, it defines a day-specific subset of "in bed time" minutes to be excluded from activity summaries computation. Default is `NULL`, i.e. no minutes excluded.
out_bed_time	A POSIXct vector. An estimated in-bed time end. Together with a corresponding entry from `in_bed_time` vector, it defines a day-specific subset of "in bed time" minutes to be excluded from activity summaries computation. Default is `NULL`, i.e. no minutes excluded.
adjust_out_colnames	A logical scalar. Whether or not to add an informative suffix to column names in the output data frame. This may happen in case any of the arguments: `subset_minutes`, or `exclude_minutes`, or `in_bed_time` and `out_bed_time` are set other than `NULL`. Default is `TRUE`.

Value

A data frame with physical activity summaries of minute level activity data. See README or vignette for summaries description.

Examples

```
## Read exemplary data
fpath_i <- system.file("extdata", extdata_fnames[1], package = "arctools")
dat_i   <- as.data.frame(data.table::fread(fpath_i))
acc     <- dat_i$vectormagnitude
acc_ts  <- lubridate::ymd_hms(dat_i$timestamp)
## Get acc data vector in "midnight_to_midnight" format
acc <- midnight_to_midnight(acc, acc_ts)
## Get wear/non-wear flag
wear_flag <- get_wear_flag(acc)
## Get valid/non-valid day flag
valid_day_flag <- get_valid_day_flag(wear_flag)
## Impute missing data in acc data vector
acc_imputed <- impute_missing_data(acc, wear_flag, valid_day_flag)

## Example 1
## Summarize PA
summarize_PA(acc, acc_ts, wear_flag, valid_day_flag)

## Example 2
## Summarize PA within minutes range corresponding to 12:00 AM - 6:00 AM
subset_12am_6am <- 1 : (6 * 1440/24)
summarize_PA(acc, acc_ts, wear_flag, valid_day_flag, subset_minutes = subset_12am_6am)

## Example 3
## Summarize PA without (i.e., excluding) minutes range corresponding to 11:00 PM - 5:00 AM.
subset_11pm_5am <- c(
  (23 * 1440/24 + 1) : 1440,   ## 11:00 PM - midnight
  1 : (5 * 1440/24)            ## midnight - 5:00 AM
)
summarize_PA(acc, acc_ts, wear_flag, valid_day_flag, exclude_minutes = subset_11pm_5am)
```

Index