Package 'mMARCH.AC'

June 18, 2026

Version 3.3.4.3

Date 2026-6-15

Title Processing of Accelerometry Data with 'GGIR' in mMARCH

Maintainer Wei Guo <wei.guo3@nih.gov>

Description Mobile Motor Activity Research Consortium for Health (mMARCH) is a collaborative network of studies of clinical and community samples that employ common clinical, biological, and digital mobile measures across involved studies. One of the main scientific goals of mMARCH sites is developing a better understanding of the inter-relationships between accelerometry-measured physical activity (PA), sleep (SL), and circadian rhythmicity (CR) and mental and physical health in children, adolescents, and adults. Currently, there is no consensus on a standard procedure for a data processing pipeline of raw accelerometry data, and few open-source tools to facilitate their development. The R package 'GGIR' is the most prominent open-source software package that offers great functionality and tremendous user flexibility to process raw accelerometry data. However, even with 'GGIR', processing done in a harmonized and reproducible fashion requires a non-trivial amount of expertise combined with a careful implementation. In addition, novel accelerometry-derived features of PA/SL/CR capturing multiscale, time-series, functional, distributional and other complimentary aspects of accelerometry data being constantly proposed and become available via non-GGIR R implementations. To address these issues, mMARCH developed a streamlined harmonized and reproducible pipeline for loading and cleaning raw accelerometry data, extracting features available through 'GGIR' as well as through non-GGIR R packages, implementing several data and feature quality checks, merging all features of PA/SL/CR together, and performing multiple analyses including Joint Individual Variation Explained (JIVE), an unsupervised machine learning dimension reduction technique that identifies latent factors capturing joint across and individual to each of three domains of PA/SL/CR. In detail, the pipeline generates all necessary R/Rmd/shell files for data processing after running 'GGIR' for accelerometer data. In module 1, all csv files in the 'GGIR' output directory were read, transformed and then merged. In module 2, the 'GGIR' output files were checked and summarized in one excel sheet. In module 3, the merged data was cleaned according to the number of valid hours on each night and the number of valid days for each subject. In module 4, the cleaned activity data was imputed by the average Euclidean norm minus one (ENMO) over all the valid days for each subject. Finally, a comprehensive report of data processing was created using Rmarkdown, and the report includes few exploratory plots and multiple commonly used features extracted from minute level actigra-

phy data. Reference: Guo W, Leroux A, Shou S, Cui L, Kang S, Strippoli MP, Preisig M, Zipunnikov V, Merikangas K (2022) Processing of accelerometry data with GGIR in Motor Activity Research Consortium for Health (mMARCH) Journal for the Measurement of Physical Behaviour, 6(1): 37-44.

URL https://github.com/WeiGuoNIMH/mMARCH.AC

BugReports https://github.com/WeiGuoNIMH/mMARCH.AC/issues

License GPL-3

Imports refund, denseFLMM, dplyr, xlsx, survival, stats, tidyr, zoo, ineq, cosinor, cosinor2, abind, minpack.lm, kableExtra, GGIR

Depends R (>= 3.6.0)

Suggests knitr, rmarkdown

Encoding UTF-8

ByteCompile true

Type Package

VignetteBuilder knitr

RoxygenNote 7.3.3

NeedsCompilation no

Author Wei Guo [aut, cre],
Andrew Leroux [aut],
Vadim Zipunnikov [aut],
Kathleen Merikangas [aut]

Repository CRAN

Date/Publication 2026-06-18 11:30:08 UTC

Contents

ActCosinor2 . . . 3
ActCosinor_long2 . . . 4
ActExtendCosinor2 . . . 5
ActExtendCosinor_long2 . . . 6
bin_data2 . . . 8
circular_hour_center . . . 8
circular_time_mean_sd . . . 9
create.shell . . . 10
data.imputation . . . 10
DataShrink . . . 12
fragmentation2 . . . 14
fragmentation_long2 . . . 15
ggir.datatransform . . . 17
ggir.summary . . . 19
IS2 . . . 20
IS_long2 . . . 21

IV2 22
IV_long2 22
jive.predict2 23
makeSleepDataMatrix 24
mMARCH.AC.maincall 24
NormalizeGGIRDate 27
PAfun 28
pheno.plot 29
RA2 30
RA_long2 30
SVDmiss2 31
Time2 32
Time_long2 32
Tvol2 33
wear_flag 34

Index **35**

`ActCosinor2` *Cosinor Model for Circadian Rhythmicity*

Description

A parametric approach to study circadian rhythmicity assuming cosinor shape.

Usage

```
ActCosinor2(x, window = 1, n1440 = 1440)
```

Arguments

`x`	`vector` vector of dimension n*1440 which reprsents n days of 1440 minute activity data
`window`	The calculation needs the window size of the data. E.g window = 1 means each epoch is in one-minute window.
`n1440`	the number of points of a day. Default is 1440 for the minute-level data.

Value

A list with elements

`mes`	MESOR which is short for midline statistics of rhythm, which is a rhythm adjusted mean. This represents mean activity level.
`amp`	amplitude, a measure of half the extend of predictable variation within a cycle. This represents the highest activity one can achieve.
`acro`	acrophase, a meaure of the time of the overall high values recurring in each cycle. Here it has a unit of radian. This represents time to reach the peak.
`acrotime`	acrophase in the unit of the time (hours)
`ndays`	Number of days modeled

References

Cornelissen, G. Cosinor-based rhythmometry. Theor Biol Med Model 11, 16 (2014). https://doi.org/10.1186/1742-4682-11-16

`ActCosinor_long2` *Cosinor Model for Circadian Rhythmicity for the Whole Dataset*

Description

A parametric approach to study circadian rhythmicity assuming cosinor shape.This function is a whole dataset wrapper for `ActCosinor`.

Usage

```
ActCosinor_long2(count.data, window = 1, daylevel = FALSE)
```

Arguments

`count.data`	`data.frame` of dimension n * (p+2) containing the p dimensional activity data for all n subject days. The first two columns have to be ID and Day. ID can be either `character` or `numeric`. Day has to be `numeric` indicating the sequence of days within each subject.
`window`	`numeric` The calculation needs the window size of the data. E.g window = 1 means each epoch is in one-minute window.
`daylevel`	`logical` If the cosinor model was run for day-level data. The default value is FALSE while the activity data for all days were used for model fitting. When the value is TRUE, the single day data were used for model fitting.

Value

A `data.frame` with the following 5 columns

`ID`	ID
`ndays`	number of days
`mes`	MESRO, which is short for midline statistics of rhythm, which is a rhythm adjusted mean. This represents mean activity level.
`amp`	amplitude, a measure of half the extend of predictable variation within a cycle. This represents the highest activity one can achieve.
`acro`	acrophase, a meaure of the time of the overall high values recurring in each cycle. Here it has a unit of radian. This represents time to reach the peak.
`acrotime`	acrophase in the unit of the time (hours)
`ndays`	Number of days modeled

`ActExtendCosinor2` *Extended Cosinor Model for Circadian Rhythmicity*

Description

Extended cosinor model based on sigmoidally transformed cosine curve using anti-logistic transformation

Usage

```
ActExtendCosinor2(
  x,
  window = 1,
  lower = c(0, 0, -1, 0, -3),
  upper = c(Inf, Inf, 1, Inf, 27),
  n1440 = 1440
)
```

Arguments

`x`	`vector` vector of dimension n*1440 which represents n days of 1440 minute activity data
`window`	The calculation needs the window size of the data. E.g window = 1 means each epoch is in one-minute window.
`lower`	A numeric vector of lower bounds on each of the five parameters (in the order of minimum, amplitude, alpha, beta, acrophase) for the NLS. If not given, the default lower bound for each parameter is set to `-Inf`.
`upper`	A numeric vector of upper bounds on each of the five parameters (in the order of minimum, amplitude, alpha, beta, acrophase) for the NLS. If not given, the default lower bound for each parameter is set to `Inf`
`n1440`	the number of points of a day. Default is 1440 for the minute-level data.

Value

A list with elements

`minimum`	Minimum value of the of the function.
`amp`	amplitude, a measure of half the extend of predictable variation within a cycle. This represents the highest activity one can achieve.
`alpha`	It determines whether the peaks of the curve are wider than the troughs: when alpha is small, the troughs are narrow and the peaks are wide; when alpha is large, the troughs are wide and the peaks are narrow.
`beta`	It dertermines whether the transformed function rises and falls more steeply than the cosine curve: large values of beta produce curves that are nearly square waves.

acrotime	acrophase is the time of day of the peak in the unit of the time (hours)
F_pseudo	Measure the improvement of the fit obtained by the non-linear estimation of the transformed cosine model
UpMesor	Time of day of switch from low to high activity. Represents the timing of the rest- activity rhythm. Lower (earlier) values indicate increase in activity earlier in the day and suggest a more advanced circadian phase.
DownMesor	Time of day of switch from high to low activity. Represents the timing of the rest-activity rhythm. Lower (earlier) values indicate decline in activity earlier in the day, suggesting a more advanced circadian phase.
MESOR	A measure analogous to the MESOR of the cosine model (or half the deflection of the curve) can be obtained from mes=min+amp/2. However, it goes through the middle of the peak, and is therefore not equal to the MESOR of the cosine model, which is the mean of the data.
ndays	Number of days modeled.

References

Marler MR, Gehrman P, Martin JL, Ancoli-Israel S. The sigmoidally transformed cosine curve: a mathematical model for circadian rhythms with symmetric non-sinusoidal shapes. Stat Med.

ActExtendCosinor_long2

Cosinor Model for Circadian Rhythmicity for the Whole Dataset

---

Description

Extended cosinor model based on sigmoidally transformed cosine curve using anti-logistic transformation.This function is a whole dataset wrapper for `ActExtendCosinor`.

Usage

```
ActExtendCosinor_long2(
  count.data,
  window = 1,
  lower = c(0, 0, -1, 0, -3),
  upper = c(Inf, Inf, 1, Inf, 27),
  daylevel = FALSE
)
```

Arguments

count.data	`data.frame` of dimension n * (p+2) containing the p dimensional activity data for all n subject days. The first two columns have to be ID and Day. ID can be either `character` or `numeric`. Day has to be `numeric` indicating the sequence of days within each subject.

window	numeric The calculation needs the window size of the data. E.g window = 1 means each epoch is in one-minute window. window size as an argument.
lower	numeric A numeric vector of lower bounds on each of the five parameters (in the order of minimum, amplitude, alpha, beta, acrophase) for the NLS. If not given, the default lower bound for each parameter is set to -Inf.
upper	numeric A numeric vector of upper bounds on each of the five parameters (in the order of minimum, amplitude, alpha, beta, acrophase) for the NLS. If not given, the default lower bound for each parameter is set to Inf
daylevel	logical If the cosinor model was run for day-level data. The default value is FALSE while the activity data for all days were used for model fitting. When the value is TRUE, the single day data were used for model fitting.

Value

A data.frame with the following 5 columns

ID	ID
ndays	number of days
minimum	Minimum value of the of the function.
amp	amplitude, a measure of half the extend of predictable variation within a cycle. This represents the highest activity one can achieve.
alpha	It determines whether the peaks of the curve are wider than the troughs: when alpha is small, the troughs are narrow and the peaks are wide; when alpha is large, the troughs are wide and the peaks are narrow.
beta	It dertermines whether the transformed function rises and falls more steeply than the cosine curve: large values of beta produce curves that are nearly square waves.
acrotime	acrophase is the time of day of the peak in the unit of the time (hours)
F_pseudo	Measure the improvement of the fit obtained by the non-linear estimation of the transformed cosine model
UpMesor	Time of day of switch from low to high activity. Represents the timing of the rest- activity rhythm. Lower (earlier) values indicate increase in activity earlier in the day and suggest a more advanced circadian phase.
DownMesor	Time of day of switch from high to low activity. Represents the timing of the rest-activity rhythm. Lower (earlier) values indicate decline in activity earlier in the day, suggesting a more advanced circadian phase.
MESOR	A measure analogous to the MESOR of the cosine model (or half the deflection of the curve) can be obtained from mes=min+amp/2. However, it goes through the middle of the peak, and is therefore not equal to the MESOR of the cosine model, which is the mean of the data.

bin_data2 *Bin data into longer windows*

Description

Bin minute level data into different time resolutions

Usage

```
bin_data2(x = x, window = 1, method = c("average", "sum"))
```

Arguments

x	vector of activity data.
window	window size used to bin the original 1440 dimensional data into. Window size should be an integer factor of 1440
method	character of "sum" or "average", function used to bin the data

Value

a vector of binned data

circular_hour_center *Center clock times on a circular 24-hour scale*

Description

Computes the shortest signed circular distance between clock times and a specified center hour on a 24-hour clock.

Usage

```
circular_hour_center(hours, center = NULL)
```

Arguments

hours	Numeric vector of hours.
center	Center hour. If NULL, circular mean is used by minimizing the total squared circular distance on a 24-hour clock.

Details

The signed difference is defined on (-12, 12] hours, so the shortest clockwise or counterclockwise distance is used.

Value

A data frame containing centered circular distances.

Examples

```
circular_hour_center(c(0:24),8)
circular_hour_center(c(NA,0:24),21)
```

`circular_time_mean_sd` *Circular mean, SD, and center adjustment for clock time*

Description

A function for calculating the circular average timing and circular standard deviation of time variables measured on a 24-hour clock. Times larger than 24 hours are wrapped onto the 24-hour scale using modulo arithmetic. The function also returns signed deviations from a specified center, or from the circular mean if center is not provided.

Usage

```
circular_time_mean_sd(times, unit_to_minute = 60, out = c("mean", "sd"))
```

Arguments

`times`	`numeric` A vector of clock times or elapsed times. Values larger than 24 hours are converted to clock time using `x %/% 24`. Missing values are ignored.
`unit_to_minute`	`numeric` The ratio of the unit of time and minute. For example, the input unit is hour, the unit2minute = 60.
`out`	`character` Specify get the mean or sd of the time variables. Default=c("mean","sd") when both mean and sd are calculated.

Details

The circular mean is computed by minimizing the total squared circular distance on a 24-hour clock. The signed difference is defined on `(-12, 12]` hours, so the shortest clockwise or counterclockwise distance is used.

Value

A list with:

`mean` Circular mean of the input times, in hours.

`sd` Circular standard deviation of the input times, in hours.

Examples

```
circular_time_mean_sd(c(1, 2, 3, 21, 20, 23))
circular_time_mean_sd(c(23, 1, 2) )
```

`create.shell` *Create a template shell script of mMARCH.AC*

Description

Create a template shell script of mMARCH.AC, named as STUDYNAME_part0.maincall.R.

Usage

```
create.shell()
```

Value

The function will create a template shell script of mMARCH.AC in the current directory, names as STUDYNAME_part0.maincall.R

`data.imputation` *Data imputation for the cleaned data with annotation*

Description

Data imputation for the merged ENMO data with annotation. The missing values were imputated by the average ENMO over all the valid days for each subject.

Usage

```
data.imputation(workdir, csvInput = NULL)
```

Arguments

`workdir`	`character` Directory where the output needs to be stored. Note that this directory must exist.
`csvInput`	`character` File name with or without directory for sample information in CSV format. The ENMO data will be read through read.csv(csvInput,header=1) command, and the missing values were imputated by the average ENMO over all the valid days for each subject at each time point. In this package, csvInput = flag_All_studyname_ENMO.data.Xs.csv. If csvInput=NULL, all available data from module 3 will be imputed.

Value

Files were written to the specified sub-directory, named as impu.flag_All_studyname_ENMO.data.Xs.csv, which Xs is the epoch size to which acceleration was averaged (seconds) in GGIR output. This excel file includs the following columns,

`filename`	accelerometer file name
`Date`	date recored from the GGIR part2.summary file
`id`	IDs recored from the GGIR part2.summary file
`calender_date`	date in the format of yyyy-mm-dd
`N.valid.hours`	number of hours with valid data recored from the part2_daysummary.csv file in the GGIR output
`N.hours`	number of hours of measurement recored from the part2_daysummary.csv file in the GGIR output
`weekday`	day of the week-Day of the week
`measurementday`	day of measurement-Day number relative to start of the measurement
`newID`	new IDs defined as the user-defined function of filename2id(), e.g. substrings of the filename
`Nmiss_c9_c31`	number of NAs from the 9th to 31th column in the part2_daysummary.csv file in the GGIR output
`missing`	"M" indicates missing for an invalid day, and "C" indicates completeness for a valid day
`Ndays`	number of days of measurement
`ith_day`	rank of the measurementday, for example, the value is 1,2,3,4,-3,-2,-1 for measurementday = 1,...,7
`Nmiss`	number of missing (invalid) days
`Nnonmiss`	number of non-missing (valid) days
`misspattern`	indicators of missing/nonmissing for all measurement days at the subject level
`RowNonWear`	number of columnns in the non-wearing matrix
`NonWearMin`	number of minutes of non-wearing
`daysleeper`	If 0 then the person is a nightsleeper (sleep period did not overlap with noon) if value=1 then the person is a daysleeper (sleep period did overlap with noon).
`remove16h7day`	indicator of a key qulity control output. If remove16h7day=1, the day need to be removed. If remove16h7day=0, the day need to be kept.
`duplicate`	If duplicate="remove", the accelerometer files will not be used in the data analysis of module5.
`ImpuMiss.b`	number of missing values on the ENMO data before imputation
`ImpuMiss.a`	number of missing values on the ENMO data after imputation
`KEEP`	The value is "keep"/"remove", e.g. KEEP="remove" if remove16h7day=1 or duplicate="remove" or ImpuMiss.a>0

DataShrink *Annotating the merged data for all accelerometer files in the GGIR output*

Description

Annotating the merged ENMO/ANGLEZ data by adding some descriptive variables such as number of valid days and missing pattern.

Usage

```
DataShrink(
  studyname,
  outputdir,
  workdir,
  QCdays.alpha = 7,
  QChours.alpha = 16,
  summaryFN = "../summary/part24daysummary.info.csv",
  epochIn = 5,
  epochOut = 60,
  useIDs.FN = NULL,
  RemoveDaySleeper = FALSE,
  trace = FALSE
)
```

Arguments

studyname	character Specify the study name that used in the output file names
outputdir	character Directory where the GGIR output was stored.
workdir	character Directory where the output needs to be stored. Note that this directory must exist.
QCdays.alpha	number Minimum required number of valid days in subject specific analysis as a quality control step in module2. Default is 7 days.
QChours.alpha	number Minimum required number of valid hours in day specific analysis as a quality control step in module2. Default is 16 hours.
summaryFN	character Filename with or without directory for sample information in CSV format, which includes summary description of each accelerometer file. Some description will be extracted and merged into the ENMO/ANGLEZ data.
epochIn	number Epoch size to which acceleration was averaged (seconds) in GGIR output. Defaut is 5 seconds.
epochOut	number Epoch size to which acceleration was averaged (seconds) in module1. Defaut is 60 seconds.

`useIDs.FN` `character` Filename with or without directory for sample information in CSV format, which inclues "filename" and "duplicate" in the headlines at least. If duplicate="remove", the accelerometer files will not be used in the data analysis of module 5-7. Defaut is NULL, which makes all accelerometer files will be used in module 5-7.

`RemoveDaySleeper` `logical` Specify if the daysleeper nights are removed from the calculation of number of valid days for each subject. Default is FALSE.

`trace` `logical` Specify if the intermediate results is printed when the function was executed. Default is FALSE.

Value

Files were written to the specified sub-directory, named as flag_ALL_studyname_ENMO.data.Xs.csv and flag_ALL_studyname_ANGLEZ.data.Xs.csv, which Xs is the epoch size to which acceleration was averaged (seconds) in GGIR output. This excel file includs the following columns,

`filename`	accelerometer file name
`Date`	date recored from the GGIR part2.summary file
`id`	IDs recored from the GGIR part2.summary file
`calender_date`	date in the format of yyyy-mm-dd
`N.valid.hours`	number of hours with valid data recored from the part2_daysummary.csv file in the GGIR output
`N.hours`	number of hours of measurement recored from the part2_daysummary.csv file in the GGIR output
`weekday`	day of the week-Day of the week
`measurementday`	day of measurement-Day number relative to start of the measurement
`newID`	new IDs defined as the user-defined function of filename2id(), e.g. substrings of the filename
`Nmiss_c9_c31`	number of NAs from the 9th to 31th column in the part2_daysummary.csv file in the GGIR output
`missing`	"M" indicates missing for an invalid day, and "C" indicates completeness for a valid day
`Ndays`	number of days of measurement
`ith_day`	rank of the measurementday, for example, the value is 1,2,3,4,-3,-2,-1 for measurementday = 1,...,7
`Nmissday`	number of missing (invalid) days
`Nnonmiss`	number of non-missing (valid) days
`misspattern`	indicators of missing/nonmissing for all measurement days at the subject level
`RowNonWear`	number of columnns in the non-wearing matrix
`NonWearMin`	number of minutes of non-wearing
`Nvalid.day`	number of valid days with/without removing daysleeper nights; It is equal to Nnonmiss when RemoveDaySleeper=FALSE.

daysleeper	If 0 then the person is a nightsleeper (sleep period did not overlap with noon) if value=1 then the person is a daysleeper (sleep period did overlap with noon) at the night. This is a night-level varialbe.
remove16h7day	indicator of a key qulity control output. If remove16h7day=1, the day need to be removed. If remove16h7day=0, the day need to be kept.
duplicate	If duplicate="remove", the accelerometer files will not be used in the data analysis of module5-7.

`fragmentation2` *Fragmentation Metrics*

Description

Fragmentation methods to study the transition between two states, e.g. sedentary v.s. active.

Usage

```
fragmentation2(
  x,
  w,
  thresh,
  bout.length = 1,
  metrics = c("mean_bout", "TP", "Gini", "power", "hazard", "all")
)
```

Arguments

x	`integer vector` of activity data.
w	`vector` of wear flag data with same dimension as `x`.
thresh	threshold to binarize the data.
bout.length	minimum duration of defining an active bout; defaults to 1.
metrics	What is the fragmentation metrics to exract. Can be "mean_bout","TP","Gini","power","hazard",or all the above metrics "all".

Details

Metrics include mean_bout (mean bout duration), TP (between states transition probability), Gini (gini index), power (alapha parameter for power law distribution) hazard (average hazard function)

Value

A list with elements

mean_r	mean sedentary bout duration
mean_a	mean active bout duration

SATP	sedentary to active transition probability
ASTP	bactive to sedentary transition probability
Gini_r	Gini index for active bout
Gini_a	Gini index for sedentary bout
h_r	hazard function for sedentary bout
h_a	hazard function for active bout
alpha_r	power law parameter for sedentary bout
alpha_a	power law parameter for active bout

References

Junrui Di, Andrew Leroux, Jacek Urbanek, Ravi Varadhan, Adam P. Spira, Jennifer Schrack, Vadim Zipunnikov. Patterns of sedentary and active time accumulation are associated with mortality in US adults: The NHANES study. bioRxiv 182337; doi: https://doi.org/10.1101/182337

Examples

```
x=c(1,2,3,4,5,6,7,8,8,5,6,7,3,2,1,1,6,6,6,6)
fragmentation2(x,w=rep(1,20),thresh=4,metrics = "all")
```

---

fragmentation_long2 *Fragmentation Metrics for Whole Dataset*

Description

Fragmentation methods to study the transition between two states, e.g. sedentary v.s. active.This function is a whole dataset wrapper for `fragmentation`

Usage

```
fragmentation_long2(
  count.data,
  weartime,
  thresh,
  bout.length = 1,
  metrics = c("mean_bout", "TP", "Gini", "power", "hazard", "all"),
  by = c("day", "subject")
)
```

Arguments

count.data	data.frame of dimension n*1442 containing the 1440 minutes of activity data for all n subject days. The first two columns have to be ID and Day. ID can be either character or numeric. Day has to be numeric indicating the sequency of days within each subject.
weartime	data.frame with dimension of count.data. The first two columns have to be ID and Day.ID can be either character or numeric. Day has to be numeric indicating the sequencey of days within each subject.
thresh	threshold to define the two states.
bout.length	minimum duration of defining an active bout; defaults to 1.
metrics	What is the fragmentation metrics to exract. Can be "mean_bout","TP","Gini","power","hazard",or all the above metrics "all".
by	Determine whether fragmentation is calcualted by day or by subjects (i.e. aggregate bouts across days). by-subject is recommended to gain more power.

Details

Metrics include mean_bout (mean bout duration), TP (between states transition probability), Gini (gini index), power (alapha parameter for power law distribution) hazard (average hazard function)

Value

A dataframe with some of the following columns

ID	identifier of the person
Day	numeric vector indicating the sequencey of days within each subject.
mean_r	mean sedentary bout duration
mean_a	mean active bout duration
SATP	sedentary to active transition probability
ASTP	bactive to sedentary transition probability
Gini_r	Gini index for active bout
Gini_a	Gini index for sedentary bout
h_r	hazard function for sedentary bout
h_a	hazard function for active bout
alpha_r	power law parameter for sedentary bout
alpha_a	power law parameter for active bout

`ggir.datatransform` *Transform the data and merge all accelerometer files in the GGIR output*

Description

An accelerometer file was transformed into wide data matrix, in which the rows represent available days and the columns including all timestamps for 24 hours. Further, the wide data was merged together.

Usage

```
ggir.datatransform(
  outputdir,
  subdir,
  studyname,
  numericID = FALSE,
  sortByid = "newID",
  f0 = 1,
  f1 = 1e+06,
  epochIn = 5,
  epochOut = 5,
  DoubleHour = c("average", "earlier", "later"),
  mergeVar = 1
)
```

Arguments

`outputdir`	`character` Directory where the GGIR output was stored.
`subdir`	`character` Sub-directory where the summary output was stored under the current directory. Defaut is "data".
`studyname`	`character` Specify the study name that used in the output file names
`numericID`	`logical` Specify if the ID is numeric when checking ID errors in module2. Default is FALSE.
`sortByid`	`character` Specify the name of "ID" for each accelerometer file in the report of module5. The value could be "newID","id" and "filename". Defaut is "filename".
`f0`	`number` File index to start with (default = 1). Index refers to the filenames sorted in increasing order.
`f1`	`number` File index to finish with. Note that file ends with the minimum of f1 and the number of files available. Default = 1000000.
`epochIn`	`number` Epoch size to which acceleration was averaged (seconds) in GGIR output. Defaut is 5 seconds.
`epochOut`	`number` Epoch size to which acceleration was averaged (seconds) in module1. Defaut is 600 seconds.

DoubleHour character Specify the method of processing the double hours for days that daylight saving time starts and ends for example. In detail, DoubleHour = c("average","earlier","later"). The acceleration data was averaged on double hours when DoulbeHour="average". Only the acceleration data in the earlier occurrence was remained for double hours while the other duplicate data were ignored when DoulbeHour="earlier". Only the acceleration data in the later occurrence was remained for double hours while the other duplicate data were ignored when DoulbeHour="later". Default is "average".

mergeVar number Specify which of the varaible need to be processed and merged. For example, mergeVar = 1 makes that the M$metalong varialbes were read from R data on the directory of /meta/basic under GGIR ourput directory, which includes "nonwearscore","clippingscore","lightmean","lightpeak","temperaturemean" and "EN". When mergeVar = 2, makes that the "enmo" and "anglez" varialbes were read from csv data on the directory of /meta/csv under GGIR ourput directory.

Value

mergeVar = 1 Six files were written to the specified sub-directory as follows,

nonwearscore_studyname_f0_f1_Xs.xlsx
Data matrix of nonwearscore, where f0 and f1 are the file index to start and finish with and Xs is the epoch size to which acceleration was averaged (seconds) in GGIR output.

clippingscore_studyname_f0_f1_Xs.xlsx
Data matrix of clippingscore

lightmean_studyname_f0_f1_Xs.xlsx
Data matrix of lightmean

lightpeak_studyname_f0_f1_Xs.xlsx
Data matrix of lightpeak

temperaturemean_studyname_f0_f1_Xs.xlsx
Data matrix of temperaturemean

EN_studyname_f0_f1_Xs.xlsx
Data matrix of EN

mergeVar = 2 Two files were written to the specified sub-directory as follows,

studyname_ENMO.dataf0_f1_Xs.xlsx
Data matrix of ENMO, where f0 and f1 are the file index to start and finish with and Xs is the epoch size to which acceleration was averaged (seconds) in GGIR output.

studyname_ANGLEZ.dataf0_f1_Xs.xlsx
Data matrix of ANGLEZ

ggir.summary *Description of all accelerometer files in the GGIR output*

Description

Description of all accelerometer files in the GGIR output and this script was executed when mode=2 in the main call.

Usage

```
ggir.summary(
  bindir = NULL,
  outputdir,
  studyname,
  numericID = FALSE,
  sortByid = "filename",
  subdir = "summary",
  part5FN = "WW_L50M125V500_T5A5",
  QChours.alpha = 16,
  filename2id = NULL,
  desiredtz = "US/Eastern",
  trace = FALSE
)
```

Arguments

bindir	character Directory where the accelerometer files are stored or list for the purpose of extracting the bin file list. Default=NULL when it is not available and therefore the bin file list is extracted from the /meta/basic folder of the GGIR output.
outputdir	character Directory where the GGIR output was stored.
studyname	character Specify the study name that used in the output file names
numericID	logical Specify if the ID is numeric when checking ID errors in module2. Default is FALSE.
sortByid	character Specify the name of "ID" for each accelerometer file in the report of module2. The value could be "newID","id" and "filename". Defaut is "filename".
subdir	character Sub-directory where the summary output was stored under the current directory. Defaut is "summary".
part5FN	character Specify which output is used in the GGIR part5 results. Defaut is "WW_L50M125V500_T5A5", which means that part5_daysummary_WW_L50M125V500_T5A5.csv and part5_personsummary_WW_L50M125V500_T5A5.csv are used in the analysis.
QChours.alpha	number Minimum required number of valid hours in day specific analysis as a quality control step in module2. Default is 16 hours.

filename2id	R function User defined function for converting filename to sample IDs. Default is NULL.
desiredtz	charcter desired timezone: see also http://en.wikipedia.org/wiki/Zone.tab. Used in g.inspectfile(). Default is "US/Eastern".
trace	logical Specify if the intermediate results is printed when the function was executed. Default is FALSE.

Value

Four files were written to the specified sub-directory

studyname_ggir_output_summary.xlsx	This excel file includs 9 pages as follows,
page 1	List of files in the GGIR output
page 2	Summary of files
page 3	List of duplicate IDs
page 4	ID errors
page 5	Number of valid days
page 6	Table of number of valid/missing days
page 7	Missing patten
page 8	Frequency of the missing pattern
page 9	Description of all accelerometer files
page 10	Inspects accelerometer file for key information, including: monitor brand, sample frequency and file header
studyname_ggir_output_summary_plot.pdf	Some plots such as the number of valid days, which were included in the module5_studyname_Data_process_report.html file as well.
part24daysummary.info.csv	Intermediate results for description of each accelerometer file.
studyname_samples_remove_temp.csv	Create studyname_samples_remove.csv file by filling "remove" in the "duplicate" column in this template. If duplicate="remove", the accelerometer files will not be used in the data analysis of module 5-7.

IS2 *Interdaily Statbility*

Description

This function calcualte interdaily stability, a nonparametric metric of circadian rhtymicity

Usage

```
IS2(x)
```

Arguments

x	data.frame of dimension ndays by p, where p is the dimension of the data.

Value

IS

References

Junrui Di et al. Joint and individual representation of domains of physical activity, sleep, and circadian rhythmicity. Statistics in Biosciences.

IS_long2 *Interdaily Statbility for the Whole Dataset*

Description

This function calcualte interdaily stability, a nonparametric metric of circadian rhtymicity. This function is a whole dataset wrapper for IS

Usage

```
IS_long2(count.data, window = 1, method = c("average", "sum"))
```

Arguments

count.data	data.frame of dimension n * (1440+2) containing the 1440 dimensional activity data for all n subject days. The first two columns have to be ID and Day. ID can be either character or numeric. Day has to be numeric indicating the sequency of days within each subject.
window	an integer indicating what is the window to bin the data before the function can be apply to the dataset. For details, see bin_data.
method	character of "sum" or "average", function used to bin the data

Value

A data.frame with the following 2 columns

ID	ID
IS	IS

References

Junrui Di et al. Joint and individual representation of domains of physical activity, sleep, and circadian rhythmicity. Statistics in Biosciences.

IV2 *Intradaily Variability*

Description

This function calcualte intradaily variability, a nonparametric metric reprsenting fragmentation of circadian rhtymicity

Usage

```
IV2(x)
```

Arguments

x	vector of activity data

Value

IV

References

Junrui Di et al. Joint and individual representation of domains of physical activity, sleep, and circadian rhythmicity. Statistics in Biosciences.

IV_long2 *Intradaily Variability for the Whole Dataset*

Description

This function calcualte intradaily variability, a nonparametric metric reprsenting fragmentation of circadian rhtymicity. This function is a whole dataset wrapper for IV.

Usage

```
IV_long2(count.data, window = 1, method = c("average", "sum"))
```

Arguments

count.data	data.frame of dimension n * (1440+2) containing the 1440 dimensional activity data for all n subject days. The first two columns have to be ID and Day. ID can be either character or numeric. Day has to be numeric indicating the sequency of days within each subject.
window	an integer indicating what is the window to bin the data before the function can be apply to the dataset. For details, see bin_data.
method	character of "sum" or "average", function used to bin the data

Value

A `data.frame` with the following 5 columns

`ID`	ID
`Day`	Day
`IV`	IV

References

Junrui Di et al. Joint and individual representation of domains of physical activity, sleep, and circadian rhythmicity. Statistics in Biosciences.

`jive.predict2` *Modified jive.predict function (package: r.jive)*

Description

Replace SVDmiss by SVDmiss2 in the function

Usage

```
jive.predict2(data.new, jive.output)
```

Arguments

`data.new`	`data.new` A list of two or more linked data matrices on which to estimate JIVE scores. These matrices must have the same column dimension N, which is assumed to be common.
`jive.output`	`jive.output` An object of class "jive", with row dimensions matching those for data.new.

Details

See jive.predict(package:r.jive) for details.

Value

See r.jive:: jive.predict for details

makeSleepDataMatrix *Make a sleep matrix based on the sleep onset and wake up time*

Description

Make a sleep matrix (sleep=1 and wake=0) based on the sleep onset and wake up time for the purpose of calculating physical acitivy features during wake up time.

Usage

```
makeSleepDataMatrix(sleepFN, epochOut = 60, impute = TRUE, outputFN)
```

Arguments

sleepFN	charcter The input file name with path of sleep onset and wake up. By default, we use part4_nightsummary_sleep_full.csv under /results/QC folder from GGIR output.
epochOut	number Epoch size to which acceleration was averaged (seconds) in part 3. Defaut is 60 seconds.
impute	logical Specify if the missing sleep time was imputed based on the averge sleep onset and wake up time. Default is TRUE.
outputFN	character The output file name that the nonsleep matrix was wrote to. It includs filename, Date, daysleeper, sleeponset, wakeup, oldDate, sleepwindow, sleepimpute, MIN1, MIN2, ..., MIN1440 for the minutes level data when flag.epochOut=60 seconds.

Value

Sleep matrix and messages of sleep data.

duplicatedays	Duplicate days of sleep data if exists
sleepproblem	Invalid sleep data if exists
sleep matrix (0/1)	write the sleep matrix to a csv file specified by outputFN

mMARCH.AC.maincall *Main Call for Data Processing after Runing GGIR for Accelerometer Data*

Description

This R script will generate all necessary R/Rmd/shell files for data processing after running GGIR for accelerometer data.

Usage

```
mMARCH.AC.maincall(
  mode,
  useIDs.FN = NULL,
  currentdir,
  studyname,
  bindir = NULL,
  outputdir,
  epochIn = 5,
  epochOut = 60,
  log.multiplier = 9250,
  use.cluster = TRUE,
  QCdays.alpha = 7,
  QChours.alpha = 16,
  QCnights.feature.alpha = c(0, 0, 0, 0),
  DoubleHour = c("average", "earlier", "later")[1],
  QC.sleepdur.avg = c(3, 12),
  QC.nblocks.sleep.avg = c(6, 29),
  Rversion = "R",
  filename2id = NULL,
  PA.threshold = c(40, 100, 400),
  PA.threshold2 = c(50, 100, 400),
  desiredtz = "US/Eastern",
  RemoveDaySleeper = FALSE,
  part5FN = "WW_L50M100V400_T5A5",
  NfileEachBundle = 20,
  holidayFN = NULL,
  trace = FALSE
)
```

Arguments

mode	number Specify which of the five modules need to be run, e.g. mode = 0 makes that all R/Rmd/sh files are generated for other modules. When mode = 1, all csv files in the GGIR output directory were read, transformed and then merged. When mode = 2, the GGIR output files were checked and summarized in one excel sheet. When mode = 3, the merged data was cleaned according to the number of valid hours on each night and the number of valid days for each subject. When mode = 4, the cleaned data was imputed.
useIDs.FN	character Filename with or without directory for sample information in CSV format, which including "filename" and "duplicate" in the headlines at least. If duplicate="remove", the accelerometer files will not be used in the data analysis of module 5-7. Defaut is NULL, which makes all accelerometer files will be used in module 5-7.
currentdir	character Directory where the output needs to be stored. Note that this directory must exist.
studyname	character Specify the study name that used in the output file names

`bindir` `character` Directory where the accelerometer files are stored or list

`outputdir` `character` Directory where the GGIR output was stored.

`epochIn` `number` Epoch size to which acceleration was averaged (seconds) in GGIR output. Defaut is 5 seconds.

`epochOut` `number` Epoch size to which acceleration was averaged (seconds) in module 3. Defaut is 60 seconds.

`log.multiplier` `number` The coefficient used in the log transformation of the ENMO data, i.e. log(log.multiplier * ENMO + 1), which have been used in module 5-7. Defaut is 9250.

`use.cluster` `logical` Specify if module1 will be done by parallel computing. Default is TRUE, and the CSV file in GGIR output will be merged for every 20 files first, and then combined for all.

`QCdays.alpha` `number` Minimum required number of valid days in subject specific analysis as a quality control step in module2. Default is 7 days.

`QChours.alpha` `number` Minimum required number of valid hours in day specific analysis as a quality control step in module2. Default is 16 hours.

`QCnights.feature.alpha`
`number` Minimum required number of valid nights in day specific mean, SD, weekday mean and weekend mean analysis as a quality control step in the JIVE analysis. Default is c(0,0,0,0), i.e. no additional data cleaning in this step.

`DoubleHour` `character` Specify the method of processing the double hours for days that daylight saving time starts and ends for example. In detail, DoubleHour = c("average","earlier","later"). The acceleration data was averaged on double hours when DoulbeHour="average". Only the acceleration data in the earlier occurrence was remained for double hours while the other duplicate data were ignored when DoulbeHour="earlier". Only the acceleration data in the later occurrence was remained for double hours while the other duplicate data were ignored when DoulbeHour="later". Default is "average".

`QC.sleepdur.avg`
`number` As taking the deault value of QC.sleepdur.avg=c(3,12), individuals were excluded with an average sleep duration <3 hour or >12 hour.

`QC.nblocks.sleep.avg`
`number` As taking the deault value of QC.nblocks.sleep.avg=c(6,29), individuals were excluded with an average number of nocturnal sleep episodes < 6 or > 29.

`Rversion` `character` R version, eg. "R/3.6.3". Default is "R".

`filename2id` `R function` User defined function for converting filename to sample IDs. Default is NULL.

`PA.threshold` `number` Threshold for light, moderate and vigorous physical activity. Default is c(40,100,400).

`PA.threshold2` `number` Second threshold for light, moderate and vigorous physical activity. Default is c(50,100,400). The activity features will end with "_C2" for those that were calculated based on PA.threshold2.

`desiredtz` `charcter` desired timezone: see also http://en.wikipedia.org/wiki/Zone.tab. Used in g.inspectfile(). Default is "US/Eastern". Used in g.inspectfile() function to inspect acceleromether file for brand, sample frequency in module 2.

`RemoveDaySleeper`
: `logical` Specify if the daysleeper nights are removed from the calculation of number of valid days for each subject. Default is FALSE.

`part5FN`
: `character` Specify which output is used in the GGIR part5 results. Defaut is "WW_L50M100V400_T5A5", which means that part5_daysummary_WW_L50M100V400_T5A5.csv and part5_personsummary_WW_L50M100V400_T5A5.csv are used in the analysis.

`NfileEachBundle`
: `number` Number of files in each bundle when the csv data were read and processed in a cluster. Default is 20.

`holidayFN`
: `character` Specify the holiday file including filename (optional), Date (mm/dd/year) and holiday (1/0) columns. When it is available, the holiday will be marked into the "weekends" group in weekday/weekend specific feature calculations in module7d. Defaut is NULL.

`trace`
: `logical` Specify if the intermediate results is printed when the function was executed. Default is FALSE.

Value

See mMARCH.AC manual for details.

`NormalizeGGIRDate` *Normalize GGIR dates to ISO format*

Description

Convert dates from heterogeneous GGIR output formats into a standard ISO 8601 date format (`"YYYY-MM-DD"`). This function performs format harmonization only and does not apply any time zone conversion.

Usage

```
NormalizeGGIRDate(x)
```

Arguments

`x`
: A vector of dates. Can be of class `character`, `factor`, `numeric`, `Date`, or `POSIXct`. Mixed formats are allowed.

Details

GGIR summary files often contain dates stored as formatted strings, Excel serial numbers, R Date numbers, or POSIX timestamps. This function detects the input type and converts all values into a consistent ISO date representation (`"YYYY-MM-DD"`), suitable for merging and aligning GGIR outputs.

Numeric values between 20000 and 60000 are treated as Excel date serials (origin `"1899-12-30"`). Other numeric values are treated as R Date values (origin `"1970-01-01"`).

Value

A character vector of dates formatted as "YYYY-MM-DD".

Examples

```
NormalizeGGIRDate("2019/01/03")
NormalizeGGIRDate("03-01-2019")
NormalizeGGIRDate(43466)        # Excel date
NormalizeGGIRDate(Sys.time())   # POSIXct
```

PAfun *Timne Metrics for Whole Dataset*

Description

This function is a whole dataset wrapper for Time

Usage

```
PAfun(count.data, weartime, PA.threshold = c(50, 100, 400))
```

Arguments

count.data	data.frame of dimension n*1442 containing the 1440 minute activity data for all n subject days. The first two columns have to be ID and Day.
weartime	data.frame with dimension of count.data. The first two columns have to be ID and Day.
PA.threshold	threshold to calculate the time in minutes of sedentary, light, moderate and vigorous activity the data.

Value

A dataframe with some of the following columns

ID	identifier of the person
Day	indicator of which day of activity it is, can be a numeric vector of sequence 1,2,... or a string of date
time	time of certain state

pheno.plot *View phenotype variables*

Description

This R script will generate plot for each variable and write description to a log file.

Usage

```
pheno.plot(
  inputFN,
  outFN = paste("plot_", inputFN, ".pdf", sep = ""),
  csv = TRUE,
  sep = " ",
  start = 3,
  read = TRUE,
  logFN = NULL,
  track = TRUE
)
```

Arguments

inputFN	character Input file name or input data
outFN	character Output pdf file name for the plots
csv	logical Specify if input file is a CSV file. Default is TRUE.
sep	character Separator between columns. Default is space. If csv=TRUE, this will not be used.
start	number The location of the first phenotype variable starts in the input file.
read	logical Specify if inputFN is a file name or a data. Default is TRUE when inputFN is a file name.
logFN	character File name of the log file. Default is NULL, while logFN=paste(inputFN,".log",sep="") in the function.
track	logical Specify if the intermediate results is printed when the function was executed. Default is TRUE.

Value

Files were written to the current directory. One is .pdf file for plots and the other is .log file for variable description.

RA2 *Relative Amplitude*

Description

This function calcualte relative amplitude, a nonparametric metric reprsenting fragmentation of circadian rhtymicity

Usage

```
RA2(x, window = 1, method = c("average", "sum"), noon2noon = FALSE)
```

Arguments

x	vector vector of activity data
window	since the caculation of M10 and L5 depends on the dimension of data, we need to include window size as an argument.
method	character of "sum" or "average", function used to bin the data
noon2noon	logical Specify if M10 and L5 were calculated from noon to noon. Default is FALSE.

Value

RA

References

Junrui Di et al. Joint and individual representation of domains of physical activity, sleep, and circadian rhythmicity. Statistics in Biosciences.

RA_long2 *Relative Amplitude for the Whole Datset*

Description

This function calcualte relative amplitude, a nonparametric metric of circadian rhtymicity. This function is a whole dataset wrapper for RA.

Usage

```
RA_long2(
  count.data,
  window = 1,
  method = c("average", "sum"),
  noon2noon = FALSE
)
```

Arguments

count.data	data.frame of dimension n * (p+2) containing the p dimensional activity data for all n subject days. The first two columns have to be ID and Day. ID can be either character or numeric. Day has to be numeric indicating the sequency of days within each subject.
window	since the caculation of M10 and L5 depends on the dimension of data, we need to include window size as an argument. This function is a whole dataset wrapper for RA.
method	character of "sum" or "average", function used to bin the data
noon2noon	logical Specify if M10 and L5 were calculated from noon to noon. Default is FALSE.

Value

A data.frame with the following 3 columns

ID	ID
Day	Day
RA	RA

SVDmiss2 *Modified SVDmiss function (package SpatioTemporal)*

Description

Modify ncomp = min(ncol(X),nrow(X),ncomp) for the matrix with nrow(X)<ncol(X)

Usage

```
SVDmiss2(X, niter = 200, ncomp = dim(X)[2], conv.reldiff = 0.001)
```

Arguments

X	X Data matrix, with missing values marked by 'NA'.
niter	niter Maximum number of iterations to run before exiting, 'Inf' will run until the 'conv.reldiff' criteria is met.
ncomp	ncomp Number of SVD components to use in the reconstruction (>0).
conv.reldiff	conv.reldiff Assume the iterative procedure has converged when the relative difference between two consecutive iterations is less than 'conv.reldiff'.

Details

See SVDmiss(package:SpatioTemporal) for details.

Value

See SpatioTemporal:: SVDmiss for details

Time2 *Time of A Certain activity State*

Description

Calculate the total time of being in certain state, e.g. sedentary, active, MVPA, etc.

Usage

```
Time2(x, w, thresh, smallerthan = TRUE, bout.length = 1)
```

Arguments

x	vector of activity data.
w	vector of wear flag data with same dimension as x.
thresh	threshold to binarize the data.
smallerthan	Find a state that is smaller than a threshold, or greater than or equal to.
bout.length	minimum duration of defining an active bout; defaults to 1.

Value

Time

Time_long2 *Timne Metrics for Whole Dataset*

Description

This function is a whole dataset wrapper for Time

Usage

```
Time_long2(count.data, weartime, thresh, smallerthan = TRUE, bout.length = 1)
```

Arguments

count.data	data.frame of dimension n*1442 containing the 1440 minute activity data for all n subject days. The first two columns have to be ID and Day.
weartime	data.frame with dimension of count.data. The first two columns have to be ID and Day.
thresh	threshold to binarize the data.
smallerthan	Find a state that is smaller than a threshold, or greater than or equal to.
bout.length	minimum duration of defining an active bout; defaults to 1.

Value

A dataframe with some of the following columns

ID	identifier of the person
Day	indicator of which day of activity it is, can be a numeric vector of sequence 1,2,... or a string of date
time	time of certain state

Tvol2 *Total Volumen of Activity for Whole Dataset*

Description

Calculate total volume of activity level, which includes TLAC (total log transfored activity counts), TAC (total activity counts).

Usage

```
Tvol2(count.data, weartime, logtransform = FALSE, log.multiplier = 9250)
```

Arguments

count.data	data.frame of dimension n*1442 containing the 1440 minute activity data for all n subject days. The first two columns have to be ID and Day.
weartime	data.frame with dimension of count.data. The first two columns have to be ID and Day.
logtransform	if TRUE, then calcualte TLAC. Or calculate TAC.
log.multiplier	number The coefficient used in the log transformation of the ENMO data, i.e. log(log.multiplier * ENMO + 1). Defaut is 9250.

Details

log transormation is defined as log(x+1).

Value

A dataframe with some of the following columns

ID	identifier of the person
Day	indicator of which day of activity it is, can be a numeric vector of sequence 1,2,... or a string of date
TAC	total activity count
TLAC	total log activity count

`wear_flag` *Create Wear/Nonwear Flags*

Description

Determine during which time period, subject should wear the device. It is preferable that user provide their own wear/non wear flag which should has the same dimension as the activity data. This function provide wear/non wear flag based on time of day.

Usage

```
wear_flag(count.data, start = "05:00", end = "23:00")
```

Arguments

`count.data`	`data.frame` of dimension n*1442 containing the 1440 minute activity data for all n subject days. The first two columns have to be ID and Day.
`start`	start time, a string in the format of 24hr, e.g. "05:00"; defaults to "05:00".
`end`	end time, a string in the format of 24hr, e.g. "23:00"; defaults to "23:00"

Details

Fragmentation metrics are usually defined when subject is awake. The `weartime` provide time periods on which those features should be extracted. This can be also used as indication of wake/sleep.

Value

A `data.frame` with same dimension and column name as the `count.data`, with 0/1 as the elments reprensting wear, nonwear respectively.

Index

ActCosinor2, 3
ActCosinor_long2, 4
ActExtendCosinor2, 5
ActExtendCosinor_long2, 6

bin_data2, 8

circular_hour_center, 8
circular_time_mean_sd, 9
create.shell, 10

data.imputation, 10
DataShrink, 12

fragmentation2, 14
fragmentation_long2, 15

ggir.datatransform, 17
ggir.summary, 19

IS2, 20
IS_long2, 21
IV2, 22
IV_long2, 22

jive.predict2, 23

makeSleepDataMatrix, 24
mMARCH.AC.maincall, 24

NormalizeGGIRDate, 27

PAfun, 28
pheno.plot, 29

RA2, 30
RA_long2, 30

SVDmiss2, 31

Time2, 32
Time_long2, 32
Tvol2, 33

wear_flag, 34