

Package ‘activpalProcessing’

December 14, 2016

Type Package

Title Process activPAL Events Files

Version 1.0.2

Date 2016-12-13

Author Kate Lyden

Maintainer Kate Lyden <activpalProcessing@gmail.com>

Description Performs estimation of physical activity and sedentary behavior variables from activPAL (PAL Technologies, Glasgow, Scotland) events files. See <<http://paltechnologies.com>> for more information on the activPAL.

License GPL-2 | GPL-3

LazyLoad yes

Depends R (>= 2.10), graphics, grDevices, stats, utils

Imports chron

LazyData yes

NeedsCompilation no

Repository CRAN

Date/Publication 2016-12-14 08:13:15

R topics documented:

activpal.file.reader	2
breaks.AP	3
events.data	4
guideline.bouts.min	4
guideline.bouts.num	5
identifyDirectory	6
identifyStudy	7
identifySubjects	7
identifyVisits	8
lit.min.AP	9
log.bed	10

log.on.off	11
log.subjects	12
mvpa.min.AP	13
on.off	14
PhysicalActivityIntervention_7_1	15
PhysicalActivityIntervention_7_2	15
PhysicalActivityIntervention_8_1	16
PhysicalActivityIntervention_8_2	17
process.AP	17
prolonged.sed.bouts.min	19
prolonged.sed.bouts.num	20
second.by.second	21
second.by.second.data	22
sed.min.AP	23
stand.min.AP	24
step.min.AP	25
wake	26

Index	27
--------------	-----------

activpal.file.reader *Read in activPAL Events File*

Description

This function reads in the events file generated by the activPAL software. Some AP events files use julian time with the origin month=12,day=30,year=1899. This function converts julian time to "%Y-%m-%d %H:%M:%S" format

Usage

```
activpal.file.reader(file.name.and.path)
```

Arguments

file.name.and.path
character string identifying the path to the file of interest

Value

A data frame with 6 columns identical to events file. Columns are renamed: 1) time=Time, 2) datacount=DataCount (samples), 3) interval=Interval (s), 4) activity=ActivityCode (0=sedentary, 1=standing, 2=stepping), 5) cumulativesteps=CumulativeStepCount, and 6) methrs=Activity Score (MET.h)

Author(s)

John Staudenmayer

Examples

```
## Not run:
activpal.file.reader("/Users/jsmith/Documents/...")
activpal.file.reader("C:/Documents and Settings/...")

## End(Not run)
```

breaks.AP

Estimate the Number of Sit to Stand Transitions

Description

This function estimates the number of sit to stand transitions. This function works on a numeric vector where 0 indicates sitting, 1 indicates standing and 2 indicates stepping. The sum of the instances AP posture transitions from 0 to 1 or 0 to 2.

Usage

```
breaks.AP(posture)
```

Arguments

posture a numeric vector (in any epoch) where 0 indicates sitting, 1 indicates standing and 2 indicates stepping.

Value

Numeric value indicating breaks from sitting

Author(s)

John Staudenmayer

Examples

```
data(second.by.second)
posture <- second.by.second.data$ap.posture

breaks.AP(posture)

# summarize by date
tapply(second.by.second.data$ap.posture, second.by.second.data$date, breaks.AP)
```

events.data *activpal events data*

Description

This data frame is an example of the data frame generated when the function `activpal.file.reader` is used to read in the `activpal events` file generated by the `activpal` software

Usage

```
data(events.data)
```

Format

A data frame with 1001 observations on the following 6 variables.

`time` a factor vector
`datacount` a integer vector
`interval` a numeric vector
`activity` a integer vector
`cumulativesteps` a integer vector
`methrs` a numeric vector

Details

Example of file generated by `activpal.file.reader` function

Examples

```
data("events.data", package="activpalProcessing")
```

guideline.bouts.min *Estimate Minutes Spent in Bouts of Activity that Qualify Towards Meeting the Physical Activity Guidelines*

Description

This function calculates time spent in guideline minutes (bouts of MVPA (≥ 3 METs) that last at least 10 min. Allows for 20

Usage

```
guideline.bouts.min(mets)
```

Arguments

mets mets is a second-by-second numeric vector

Value

numeric value indicating number of minutes spent in guideline bouts

Author(s)

John Staudenmayer

See Also

[guideline.bouts.num](#)

Examples

```
data(second.by.second)
mets <- second.by.second.data$mets

guideline.bouts.min(mets)

# summarize by date
tapply(second.by.second.data$mets, second.by.second.data$date, guideline.bouts.min)
```

guideline.bouts.num *Estimate the Number of Bouts of Activity that Qualify Towards Meeting the Physical Activity Guidelines*

Description

This function calculates the number of guideline bouts (bouts of MVPA (≥ 3 METs) that last at least 10 min. Allows for 20 percent of time below 3 METs i.e. if bout is 10 min long, 2 min can be < 3 METs

Usage

```
guideline.bouts.num(mets)
```

Arguments

mets mets is a second-by-second numeric vector

Value

numeric value indicating the number of guideline bouts

Author(s)

John Staudenmayer

See Also

[guideline.bouts.min](#)

Examples

```
data(second.by.second)
mets <- second.by.second.data$mets

guideline.bouts.num(mets)

# summarize by date
tapply(second.by.second.data$mets, second.by.second.data$date, guideline.bouts.num)
```

identifyDirectory *Identify Working Directory*

Description

This function identifies the path to the folder where data files and logs are saved. Results files will also be saved to this folder

Usage

```
identifyDirectory(path)
```

Arguments

path A character string identifying the path to the working directory

Value

A character string identifying the path to the working directory

Author(s)

Kate Lyden

Examples

```
## Not run:
identifyDirectory("/Users/jsmith/Documents/...")
identifyDirectory("C:/Documents and Settings/...")

## End(Not run)
```

identifyStudy	<i>Identify Study Name</i>
---------------	----------------------------

Description

This function identifies the study name. This function helps ensure the correct data are being processed. Study name is identified from the log of subjects.

Usage

```
identifyStudy(directory, name.of.log.subjects)
```

Arguments

directory	Directory is a character string identifying the path to the working directory. default value is directory
name.of.log.subjects	character string identifying the name of the log of subjects

Value

character string identifying study name

Author(s)

Kate Lyden

Examples

```
## Not run:  
identifyStudy("/Users/jsmith/Documents...log.subjects.csv")  
  
## End(Not run)
```

identifySubjects	<i>Identify Subjects in Study</i>
------------------	-----------------------------------

Description

This function identifies subject that have AP files to be processed

Usage

```
identifySubjects(directory=directory, name.of.log.subjects)
```

Arguments

directory Directory is a character string identifying the path to the working directory. Default value is directory

name.of.log.subjects
character string identifying the name of the log of subjects

Value

A vector of subject IDs

Author(s)

Kate Lyden

Examples

```
## Not run:
identify.study("/Users/jsmith/Documents...log.subjects")

## End(Not run)
```

identifyVisits

Identify Visits in Study

Description

This function identifies visits that have AP files to be processed

Usage

```
identifyVisits(directory=directory, name.of.log.subjects)
```

Arguments

directory Directory is a character string identifying the path to the working directory. Default value is directory

name.of.log.subjects
character string identifying the name of the log of subjects

Value

A vector of visit names

Author(s)

Kate Lyden

Examples

```
## Not run:
identifyVisits("/Users/jsmith/Documents/...log.subjects.csv")

## End(Not run)
```

`lit.min.AP`*Estimate Time in Light Intensity Activity*

Description

This function estimates time spent in light intensity activity. Light intensity activity is activity in the standing or stepping posture and between 1.5-2.99 METs.

Usage

```
lit.min.AP(mets, posture, epoch = 1)
```

Arguments

<code>mets</code>	<code>mets</code> is a vector of met values
<code>posture</code>	<code>posture</code> is a vector of values 0-2 indicating 0=sitting, 1=standing, 2=stepping
<code>epoch</code>	numeric value indicating what epoch (in seconds) the data are in (e.g. 60 = 1 minute epochs). The default value is 1 second

Details

The default settings of the AP assign a met value of 1.25 to sitting, 1.4 to standing and estimates a met value for stepping activities. Cadence is used to estimate mets for stepping. Although the default met value for standing is 1.4, which is below the traditional met threshold for light intensity, standing is always considered light because it is an upright posture and only seated activities are considered sedentary

Value

a numeric value indicating hours spent in light intensity activity

Author(s)

Kate Lyden

See Also

[sed.min.AP](#) [mvpa.min.AP](#)

Examples

```
data(second.by.second.data)
mets <- second.by.second.data$mets
posture <- second.by.second.data$ap.posture
lit.min.AP(mets,posture,epoch=1)
```

log.bed

Example of a Bed Log

Description

This data frame is an example of a bed log. Bed logs need to be in this format to work with this package. If no bed log is provided all wear time will be included in analysis

Usage

```
data(log.bed)
```

Format

A data frame with 32 observations on the following 15 variables.

id a numeric vector

visit a factor with levels visit 2 visit1 visit2

study a factor with levels PhysicalActivityIntervention

date.out.month a numeric vector

date.out.day a numeric vector

date.out.year a numeric vector

time.out.hour a numeric vector

time.out.minute a numeric vector

time.out.seconds a numeric vector

date.in.month a numeric vector

date.in.day a numeric vector

date.in.year a numeric vector

time.in.hour a numeric vector

time.in.minute a numeric vector

time.in.seconds a numeric vector

Details

Example bed log. Data are made up for 2 subjects each (id=7 & 8) and 2 visits (visit= 1 & 2)

Examples

```
# load example bed log
data("log.bed",package="activpalProcessing")
```

log.on.off

Example on/off Log

Description

This data frame is an example of an on/off log. If no on/off log is provided all AP data will be included in analysis

Usage

```
data(log.on.off)
```

Format

A data frame with 59 observations on the following 15 variables.

id a numeric vector

visit a factor with levels visit1 visit2

study a factor with levels PhysicalActivityIntervention

date.on.month a numeric vector

date.on.day a numeric vector

date.on.year a numeric vector

time.on.hour a numeric vector

time.on.minute a numeric vector

time.on.seconds a numeric vector

date.off.month a numeric vector

date.off.day a numeric vector

date.off.year a numeric vector

time.off.hour a numeric vector

time.off.minute a numeric vector

time.off.seconds a numeric vector

Details

Example bed log. Data are made up for 2 subjects and 2 visits each

Examples

```
# load example on.off log
data("log.on.off", package="activpalProcessing")
```

log.subjects	<i>Example Log of Subjects</i>
--------------	--------------------------------

Description

This data frame provides a list of all subjects and visits that have AP data to be processed. Also provides the name of the study of interest

Usage

```
data(log.subjects)
```

Format

A data frame with 4 observations on the following 3 variables.

id a numeric vector

visit a factor with levels visit1 visit2

study a factor with levels PhysicalActivityIntervention

Details

Example bed log. Data are made up for 2 subjects and 2 visits each

Examples

```
# load example subject log
data("log.subjects", package="activpalProcessing")
```

`mvpa.min.AP`*Estimate Time in Moderate-to-Vigorous Intensity Activity*

Description

This function estimates time spent in moderate-to-vigorous intensity activity (MVPA). MVPA is activity ≥ 3 METs.

Usage

```
mvpa.min.AP(mets, epoch)
```

Arguments

<code>mets</code>	mets is a vector of met values
<code>epoch</code>	numeric value indicating what epoch (in seconds) the data are in (e.g. 60 = 1 minute epochs). The default value is 1 second

Value

a numeric value indicating hours spent in mvpa

Author(s)

Kate Lyden

See Also

[sed.min.AP](#) [lit.min.AP](#)

Examples

```
data(second.by.second)
mets <- second.by.second.data$mets

mvpa.min.AP(mets, epoch=1)

# summarize by date
tapply(second.by.second.data$mets, second.by.second.data$date, mvpa.min.AP, epoch=1)
```

on.off	<i>Read in on/off log and Identify on/off Times in Second-by-Second AP Events File</i>
--------	--

Description

This function reads in on/off log and marks on/off time in sec-by-second file. On/off log must be in correct format. A column named "off" will be added to the second-by-second file. 1 indicates the device is off and 0 indicates the device is on.

Usage

```
on.off(directory = directory, id, visit, name.of.log.on.off, data)
```

Arguments

directory	Directory is a character string identifying the path to the working directory. Default value is directory
id	Character string identifying the id of the subject of interest. This argument should correspond to the id used in the on/off log
visit	Character string identifying the visit of interest for the subject. This argument should correspond to the visit identifier used in the on/off log
name.of.log.on.off	Character string identifying the name of the on/off log
data	data is the second-by-second AP events file

Value

A data frame with the column "off" added to the sec-by-sec AP events file

Note

Follow the example on/off log provided with this package (log.on.off)

Author(s)

Kate Lyden

Examples

```
## Not run:
on.off(directory, log.on.off, data)

## End(Not run)
```

PhysicalActivityIntervention_7_1
Example AP Events File

Description

This data frame is an example AP events file (as downloaded from the AP software)

Usage

```
data(PhysicalActivityIntervention_7_1)
```

Format

A data frame with 40386 observations on the following 6 variables.

Time a numeric vector

DataCount..samples. a numeric vector

Interval..s. a numeric vector

ActivityCode..0.sedentary..1..standing..2.stepping. a numeric vector

CumulativeStepCount a numeric vector

Activity.Score..MET.h. a numeric vector

Examples

```
data(PhysicalActivityIntervention_7_1)
```

PhysicalActivityIntervention_7_2
Example AP Events File

Description

This data frame is an example AP events file (as downloaded from the AP software).

Usage

```
data(PhysicalActivityIntervention_7_2)
```

Format

A data frame with 41824 observations on the following 6 variables.

Time a numeric vector

DataCount..samples. a numeric vector

Interval..s. a numeric vector

ActivityCode..0.sedentary..1..standing..2.stepping. a numeric vector

CumulativeStepCount a numeric vector

Activity.Score..MET.h. a numeric vector

Examples

```
data(PhysicalActivityIntervention_7_2)
```

```
PhysicalActivityIntervention_8_1  
  Example AP Events File
```

Description

This data frame is an example AP events file (as downloaded from the AP software).

Usage

```
data(PhysicalActivityIntervention_8_1)
```

Format

A data frame with 60113 observations on the following 6 variables.

Time a numeric vector

DataCount..samples. a numeric vector

Interval..s. a numeric vector

ActivityCode..0.sedentary..1..standing..2.stepping. a numeric vector

CumulativeStepCount a numeric vector

Activity.Score..MET.h. a numeric vector

Examples

```
data(PhysicalActivityIntervention_8_1)
```

PhysicalActivityIntervention_8_2
Example AP Events File

Description

This data frame is an example AP events file (as downloaded from the AP software).

Usage

```
data(PhysicalActivityIntervention_8_2)
```

Format

A data frame with 45991 observations on the following 6 variables.

Time a numeric vector

DataCount..samples. a numeric vector

Interval..s. a numeric vector

ActivityCode..0.sedentary..1..standing..2.stepping. a numeric vector

CumulativeStepCount a numeric vector

Activity.Score..MET.h. a numeric vector

Examples

```
data(PhysicalActivityIntervention_8_2)
```

process.AP *Process AP Data*

Description

This function processes multiple AP events files for multiple subjects/visits in a given study. This function uses all other functions included in the package "process.AP" to read in all AP events files saved in a given directory and estimate physical activity and sedentary behavior variables. Three files are created and saved to the specified directory 1) Sleep Wake Wear Table, 2) Results Table and 3) Mean Table

Usage

```
process.AP(directory, name.of.log.subjects, name.of.log.bed, name.of.log.on.off)
```

Arguments

directory	character string identifying the path to the working directory.
name.of.log.subjects	character string identifying the name of the log of subjects
name.of.log.bed	character string identifying the name of the bed log. Default value is NULL
name.of.log.on.off	character string identifying the name of the on/off log. Default value is NULL

Details

AP events files must be saved using the following naming convention - "study name_subject id_visit.csv". For example, if study name = "Physical Activity Intervention", subject id = 1, visit = 3 the events file is named "Physical Activity Intervention_1_3.csv". bed log and on/off log do not need to be provided. Results will be estimated from all AP data (e.g. sleep time and off time will be included in analyses) - results and sleep wake wear table will indicate no valid logs were provided. If logs are provided, they must be in the exact format of the example logs provided with this package. Example logs may be loaded into R and exported to the specified working directory (see Examples). These files should serve as templates for the user to create their own logs.

Value

Results Table - A .csv file generated and saved in the specified directory. The file contains the following variables estimated per day device was worn. 1) study - name of study 2) sub - subject ID 3) visit - visit ID 4) date - date of wear 5) hours.awake.worn - number of hours subject was awake and wearing the device 6) met.hours - total met hours 7) step.count - total step count 8) sed.mins - number of minutes spent sitting 9) stand.mins - number of minutes spent standing 10) step.mins - number of minutes spent stepping 11) lit.mins - number of minutes spent in light intensity (1.5-2.99) physical activity 12) mvpa.mins - number of minutes spent in moderate-to-vigorous intensity (≥ 3 METs) activity 13) breaks - number of breaks from sedentary time 14) break.rate - breaks per sedentary hour 15) guideline.minutes - number of minutes spent in bouts of activity that qualify towards meeting the physical activity guidelines (≥ 3 METs and at least 10 minutes in duration) 16) num.guideline bout - number of bouts of activity that qualify towards meeting the physical activity guidelines (≥ 3 METs and at least 10 minutes in duration) 17) min.in.sed.30 - number of minutes spent in sedentary bouts at least 30 minutes in duration 18) min.in.sed.60 - number of minutes spent in sedentary bouts at least 60 minutes in duration 19) percent.of.hours.awake.worn.sed - percent of waking hours sedentary 20) percent.of.hours.awake.worn.lit - percent of waking hours in light intensity activity 21) percent.of.hours.awake.worn.mvpa - percent of waking hours in mvpa

Sleep Wake Wear Table - A .csv file generated and saved in the specified directory. The file contains the following variables estimated per day device was worn. If valid bed and/or on/off logs are not provided sleep wake wear table will indicate that the logs are missing. 1) study - name of study 2) sub - subject ID 3) visit - visit ID 4) date - date of wear 5) awake.hours - total hours spent awake 6) total.sleep.hours - total hours spent asleep 7) total.wear.hours - total hours device was worn 8) non.wear.hours - total hours device was not worn 9) hours.awake.worn - total hours the device was worn while awake 10) hours.awake.not.worn - total hours the device was not worn while awake 11) hours.sleep.worn - total hours the device was worn while asleep 12) hours.sleep.not.worn - total hours the device was not worn while asleep

Means Table - A .csv file generated and saved in the specified directory. The file summarizes (average, sd and 95 percent CI) the variables in Results Table by subject and visit

Author(s)

Kate Lyden

Examples

```
## Not run:

# identify working directory
directory <- identifyDirectory("/Users/jsmith/Documents/InterventionStudy/")

# load example logs and export a tempaltes to working directory
data("log.subjects",package="activpalProcessing")
data("log.bed",package="activpalProcessing")
data("log.on.off",package="activpalProcessing")

write.table(log.subjects,file=paste(directory,"temp.log.subjects.csv",sep=""),sep=" ",row.names=F)
write.table(log.bed,file=paste(directory,"temp.log.bed.csv",sep=""),sep=" ",row.names=F)
write.table(log.on.off,file=paste(directory,"temp.log.on.off.csv",sep=""),sep=" ",row.names=F)

# process AP events files saved in directory
process.AP(directory=directory, name.of.log.subjects, name.of.log.bed, name.of.log.on.off)

## End(Not run)
```

prolonged.sed.bouts.min

Estimate Time in Prolonged Sedentary Bouts

Description

This function calculates the minutes spent in sitting/lying bouts that last at least 'n' minutes in duration. User specifies n.

Usage

```
prolonged.sed.bouts.min(posture, epoch = 1, n)
```

Arguments

posture	posture is a vector indicating sitting (0), standing (1) and stepping (2)
epoch	numeric value indicating what epoch (in seconds) the data are in (e.g. 60 = 1 minute epochs). The default value is 1 second
n	n is a numeric value indicating the duration (in minutes) of prolonged sedentary bouts of interest

Value

A numeric value indicating minutes spent in sedentary bouts longer than 'n' minutes in duration

Author(s)

John Staudenmayer

See Also

[prolonged.sed.bouts.num](#)

Examples

```
data(second.by.second.data)
posture <- second.by.second.data$ap.posture

# estimates minutes spent in sedentary bouts lasting at least 60 minutes
prolonged.sed.bouts.min(posture, epoch=1, n=60)
```

prolonged.sed.bouts.num

Estimate The Number of Prolonged Sedentary Bouts

Description

This function calculates the number of sitting/lying bouts that last at least 'n' minutes in duration. User specifies n.

Usage

```
prolonged.sed.bouts.num(posture, epoch = 1, n)
```

Arguments

posture	posture is vector indicating sitting (0), standing (1) and stepping (2).
epoch	numeric value indicating what epoch (in seconds) the data are in (e.g. 60 = 1 minute epochs). The default value is 1 second
n	n is a numeric value indicating the duration (in minutes) of prolonged sedentary bouts of interest

Value

A numeric value indicating the number of sedentary bouts longer than 'n' minutes in duration

Author(s)

John Staudenmayer

See Also

[prolonged.sed.bouts.min](#)

Examples

```
data(second.by.second.data)
posture <- second.by.second.data$ap.posture

# estimate number of sedentary bouts lasting at least 60 minutes
prolonged.sed.bouts.num(posture,epoch=1,n=60)
```

second.by.second

Convert AP Events File to a Second-By-Second File

Description

This function converts the AP Events file to a second-by-second data frame. The AP Events file must be a data frame in the format produced by the function `activpal.file.reader`.

Usage

```
second.by.second(data)
```

Arguments

`data` Data is a data frame in the format produced by the function `activpal.file.reader`.

Details

This function works on AP Events files read in by `activpal.file.reader`.

Value

data frame with 9 variables in second-by-second epochs: 1) `time` = date and time stamps beginning at start of recording and ending at last recording 2) `date` = date stamp beginning at first day of recording and ending at last day of recording 3) `ap.posture` = posture 0 = sitting, 1 = standing, 3 = stepping 4) `mets` = met value 5) `met.hours` = met value expressed as met.hour (e.g. `mets/3600`) 6) `steps` = cumulative step count 7) `fifteen.sec.mets` = 15 second epoch met value (e.g. average met value over 15 seconds) 8) `one.min.mets` = 60 second epoch met value (e.g. average met value over 60 seconds) 9) `num.events` = numeric value indicating which event this second of data corresponds to

Author(s)

John Staudenmayer, Kate Lyden

See Also

[activpal.file.reader.julian](#)

Examples

```
## Not run:  
data <- activpal.file.reader(file.name.and.path)  
  
second.by.second(data)  
  
## End(Not run)  
data(events.data)  
second.by.second(events.data)
```

second.by.second.data *Second-by-second data frame*

Description

This data frame is an example of the data frame generated when the function `second.by.second` is applied to an `activpal` events file

Usage

```
data(second.by.second.data)
```

Format

A data frame with 696444 observations on the following 9 variables.

```
time a factor vector  
date a factor vector  
ap.posture a integer vector  
mets a numeric vector  
met.hours a numeric vector  
steps a integer vector  
fifteen.sec.mets a numeric vector  
one.min.mets a numeric vector  
num.events a integer vector
```

Details

Example of file generated by second.by.second function

Examples

```
# load data
data("second.by.second.data", package="activpalProcessing")
```

sed.min.AP	<i>Estimate Time in Sedentary</i>
------------	-----------------------------------

Description

This function to calculates the minutes spent sitting/lying.

Usage

```
sed.min.AP(posture, epoch = 1)
```

Arguments

posture	posture is a vector indicating sitting (0), standing (1) and stepping (2)
epoch	numeric value indicating what epoch (in seconds) the data are in (e.g. 60 = 1 minute epochs). The default value is 1 second

Details

The default settings of the AP assign a met value of 1.25 to sitting, 1.4 to standing and estimates a met value for stepping activities. Cadence is used to estimate mets for stepping. With these settings all standing activity will be considered sedentary (e.g. < 1.5 mets). In the second-by-second function in this package standing met values are changed to 1.5 so that standing is considered a light intensity activity. Thus the only events considered sedentary are sitting/lying events

Value

a numeric value indicating minutes spent sedentary

Author(s)

Kate Lyden

See Also

[lit.min.AP](#) [mvpa.min.AP](#) [stand.min.AP](#) [step.min.AP](#)

Examples

```
data(second.by.second.data)
posture <- second.by.second.data$ap.posture

sed.min.AP(posture,epoch=1)

# summarize by date
tapply(second.by.second.data$ap.posture, second.by.second.data$date, sed.min.AP,epoch=1)
```

stand.min.AP	<i>Estimate Time Spent Standing</i>
--------------	-------------------------------------

Description

This function to calculates the hours spent standing

Usage

```
stand.min.AP(posture, epoch = 1)
```

Arguments

posture	posture is vector indicating sitting (0), standing (1) and stepping (2)
epoch	numeric value indicating what epoch (in seconds) the data are in (e.g. 60 = 1 minute epochs). The default value is 1 second

Value

a numeric value indicating hours spent standing

Author(s)

Kate Lyden

See Also

[sed.min.AP](#) [step.min.AP](#)

Examples

```
data(second.by.second.data)
posture <- second.by.second.data$ap.posture

stand.min.AP(posture,epoch=1)

# summarize by date
tapply(second.by.second.data$ap.posture, second.by.second.data$date, stand.min.AP,epoch=1)
```

`step.min.AP`*Estimate Time Spent Stepping*

Description

This function to calculates the hours spent stepping

Usage

```
step.min.AP(posture, epoch = 1)
```

Arguments

posture	posture is a vector indicating sitting (0), standing (1) and stepping (2)
epoch	numeric value indicating what epoch (in seconds) the data are in (e.g. 60 = 1 minute epochs). The default value is 1 second

Value

a numeric value indicating hours spent stepping

Author(s)

Kate Lyden

See Also

[sed.min.AP](#) [stand.min.AP](#)

Examples

```
data(second.by.second.data)
posture <- second.by.second.data$ap.posture

step.min.AP(posture, epoch=1)

# summarize by date
tapply(second.by.second.data$ap.posture, second.by.second.data$date, step.min.AP, epoch=1)
```

wake	<i>Read in bed log and Identify Wake/Sleep Times in Second-by-Second AP Events File</i>
------	---

Description

This function reads in bed log and marks wake/sleep time in sec-by-second file. Bed log must be in correct format. A column named "wake" will be added to the second-by-second file. 1 indicates the participant is awake and 0 indicates the participant is sleeping.

Usage

```
wake(directory = directory, id, visit, name.of.log.bed, data)
```

Arguments

directory	Directory is a character string identifying the path to the working directory. Default value is directory
name.of.log.bed	Character string identifying the name of the bed log
id	Character string identifying the id of the subject of interest. This argument should correspond to the id used in the bed log
visit	Character string identifying the visit of interest for the subject. This argument should correspond to the visit identifier used in the bed log
data	data is the second-by-second AP events file

Value

A data frame with the column "wake" added to the sec-by-sec AP events file

Note

Follow the example bed log provided with this package (log.bed)

Author(s)

Kate Lyden

Examples

```
## Not run:
on.off(directory, id, visit, log.bed, data)

## End(Not run)
```

Index

- *Topic **activity**
 - guideline.bouts.min, 4
 - guideline.bouts.num, 5
 - mvpa.min.AP, 13
 - *Topic **activpal**
 - activpal.file.reader, 2
 - *Topic **bed**
 - wake, 26
 - *Topic **breaks**
 - breaks.AP, 3
 - *Topic **datasets**
 - events.data, 4
 - log.bed, 10
 - log.on.off, 11
 - log.subjects, 12
 - PhysicalActivityIntervention_7_1, 15
 - PhysicalActivityIntervention_7_2, 15
 - PhysicalActivityIntervention_8_1, 16
 - PhysicalActivityIntervention_8_2, 17
 - second.by.second.data, 22
 - *Topic **events**
 - activpal.file.reader, 2
 - *Topic **extrapolate**
 - second.by.second, 21
 - *Topic **library**
 - identifyDirectory, 6
 - *Topic **light**
 - lit.min.AP, 9
 - *Topic **loop**
 - process.AP, 17
 - *Topic **mvpa**
 - guideline.bouts.min, 4
 - guideline.bouts.num, 5
 - *Topic **off time**
 - on.off, 14
 - *Topic **process**
 - process.AP, 17
 - *Topic **sedentary**
 - prolonged.sed.bouts.num, 20
 - sed.min.AP, 23
 - *Topic **sitting**
 - prolonged.sed.bouts.min, 19
 - prolonged.sed.bouts.num, 20
 - *Topic **sit**
 - sed.min.AP, 23
 - *Topic **sleep**
 - wake, 26
 - *Topic **steps**
 - step.min.AP, 25
 - *Topic **study**
 - identifyStudy, 7
 - *Topic **subject**
 - identifySubjects, 7
 - *Topic **transitions**
 - breaks.AP, 3
 - *Topic **upright**
 - stand.min.AP, 24
 - *Topic **visits**
 - identifyVisits, 8
 - *Topic **wear time**
 - on.off, 14
- activpal.file.reader, 2, 22
- breaks.AP, 3
- events.data, 4
- guideline.bouts.min, 4, 6
- guideline.bouts.num, 5, 5
- identifyDirectory, 6
- identifyStudy, 7
- identifySubjects, 7
- identifyVisits, 8

julian, [22](#)

lit.min.AP, [9](#), [13](#), [23](#)
log.bed, [10](#)
log.on.off, [11](#)
log.subjects, [12](#)

mvpa.min.AP, [9](#), [13](#), [23](#)

on.off, [14](#)

PhysicalActivityIntervention_7_1, [15](#)
PhysicalActivityIntervention_7_2, [15](#)
PhysicalActivityIntervention_8_1, [16](#)
PhysicalActivityIntervention_8_2, [17](#)
process.AP, [17](#)
prolonged.sed.bouts.min, [19](#), [21](#)
prolonged.sed.bouts.num, [20](#), [20](#)

second.by.second, [21](#)
second.by.second.data, [22](#)
sed.min.AP, [9](#), [13](#), [23](#), [24](#), [25](#)
stand.min.AP, [23](#), [24](#), [25](#)
step.min.AP, [23](#), [24](#), [25](#)

wake, [26](#)