Acelerometria

July 12, 2018

`connectOverDistance`	*Connect intervals that are close enough Generates a smaller dataframe of connected intervals using a data frame of given intervals and a distance that allows tow of them to be connected*

Description

Connect intervals that are close enough Generates a smaller dataframe of connected intervals using a data frame of given intervals and a distance that allows tow of them to be connected

Usage

```
connectOverDistance(interval, distance = dminutes(30))
```

Arguments

`interval`	Initial dataframe of intervals to be connected
`distance`	distance between ttwo intervals to allo the connection of both in just one.

Value

A dataframe of intervals, having less or equal rows tan the original

connectOverInterval — *Connect intervals of a first dataframe using a second dataframe of intervals*

Description

Connect the intervals of a first dataframe given that the can be considered connected if the separation between two of them are covered by a interval of a second dataframe.

Usage

```
connectOverInterval(interval1, interval2)
```

Arguments

interval1	first dataframe
interval2	second dataframe

Value

A dataframe of intervals representing the connected intervals

criterioBout — *Indicates which rows meets the criteria for be considered part of a Bout meeting certain criteria*

Description

Generates a vector that indicates for each row of a dataframe (usually epoch or BIN file) if that row meets the criteria to be considered part of a Bout (veryfy a criteria for ca percent of time of a minimum duration),

Usage

```
criterioBout(df, pctBouts = 1, durBoutMin = dseconds(5),
  durEpoch = dseconds(5))
```

Arguments

df	data frame with columns .criterio and eventually .criterioNW (that represents NonWear time as TRUE/FALSE)
durBoutMin	minimum amount of time that the condition muest be met to be aconsidered a Bout
durEpoch	amount of time that represents each row of the dataframe (duration of a epoch usually)
pctBout	represents fraction of time that the .criterio must be TRUE

Value

a boolean vector (TRUE/FAlSE) indicating if the condition of belonging to a Bout is met.

`criterioENMO` *Indicates which rows meets certain limits for ENMO*

Description

Generates a vector that indicates for each row of a dataframe (usually epoch or BIN file) if that row verifies the condition limInf<=df[["ENMO"]] <=limSup

Usage

```
criterioENMO(df, limInf = 0, limSup = Inf)
```

Arguments

`df`	data frame with columns ENMO and eventually .criterioNW (that represents NonWear time as TRUE/FALSE)
`limInf`	inferior limit for ENMO
`limSup`	superior limit for ENMO

Value

a boolean vector (TRUE/FAlSE) indicating if the condition is met.

`criterioSIB` *Indicates which rows meets the criteria for be considered part of a SIB (Sustained Inactivity Bouts)*

Description

Generates a vector that indicates for each row of a dataframe (usually epoch or BIN file) if that row verifies the condition to be considered a SIB, i.e Anglez variying a few degrees for a certain amount of time (durBoutMin, and ENMO lower than a low value)

Usage

```
criterioSIB(df, critAnglez = 5, limSup = 25/1000,
  durBoutMin = dminutes(5), durEpoch = dseconds(5))
```

Arguments

df	data frame with columns ANGLEZ, ENMO and eventually .criterioNW (that represents NonWear time as TRUE/FALSE)
critAnglez	represents maximum of deviation (in both directios) of angle Z that it is allowed during a SIB
limSup	superior limit for ENMO
durBoutMin	minimum amount of time that the conditions must be met to beconsidered a SIB period
durEpoch	amount of time that represents each row of the dataframe (duration of a epoch usually)

Value

a boolean vector (TRUE/FAlSE) indicating if the condition of belonging to a SIB is met.

---

getInterval *Convert a data frame with a column called .criterio to a data frame of intervals of bouts of that criteria*

Description

data frame of intervals of Bouts meeting certain criteria,

Usage

```
getInterval(df, pctBouts = 1, durBoutMin = dseconds(5),
  durEpoch = dseconds(5), units = "mins")
```

Arguments

df	data frame with columns .criterio
durBoutMin	minimum amount of time that the condition muest be met to be aconsidered a Bout
durEpoch	amount of time that represents each row of the dataframe (duration of a epoch usually)
units	Units of time to show certain summaries. Une of c("secs","mins","hours","days")
pctBout	represents fraction of time that the .criterio must be TRUE

Value

a list with data frame of intervals and certain summaries.

`getSummary` *Create a summery for the column .criterio of a data frame representing accelerometer data*

Description

Generates a summary for a epoch or BIN file

Generates a summary for a epoch or BIN file

Usage

```
getSummary(df, offset = dhours(0), minimoHorasValidas = 20,
  maximoHorasNonWear = 2)

getSummary(df, offset = dhours(0), minimoHorasValidas = 20,
  maximoHorasNonWear = 2)
```

Arguments

`df`	data frame with columns timestamp and .criterio
`offset`	Will add a quantity to the timestamp to assign the instant of time to the appropiate day
`minimoHorasValidas`	explicar
`maximoHorasNonWear`	explicar
`Variable`	minimum amount of time that the condition muest be met to be aconsidered a Bout
`durEpoch`	amount of time that represents each row of the dataframe (duration of a epoch usually)
`df`	data frame with columns timestamp and .criterio
`offset`	Will add a quantity to the timestamp to assign the instant of time to the appropiate day
`Variable`	minimum amount of time that the condition muest be met to be aconsidered a Bout
`durEpoch`	amount of time that represents each row of the dataframe (duration of a epoch usually)
`minimoHorasValidas`	explicar
`maximoHorasNonWear`	explicar

Value

a list with a summary. This is the info that we use to define activity variables on a daily basis and on a global value.

a list with a summary. This is the info that we use to define activity variables on a daily basis and on a global value.

`interval2criterio` *Convert a dataframe of intervals, given a vector of timestamps representing epochs to a boolean vector representing if that epoch belong to one of the intervals*

Description

Convert a dataframe of intervals, given a vector of timestamps representing epochs to a boolean vector representing if that epoch belong to one of the intervals

Usage

```
interval2criterio(ts, intervalos, durEpoch = dseconds(5))
```

Arguments

`ts`	vector of timestamps
`intervalos`	dataframe of intervals, with columns `to` and {from

Value

a logical vector of length(ts) indicating if that time velong to a interval

---

`intervalBED` *Obtain the intervals formed by the best candidates every day to be considered as time and sleeping in bed*

Description

For each day of accelerometry recorded choose an interval as best candidate to be considered as interval in bed, and inside of it, mark another interval as time from first sleep to awake before going out of bed

Usage

```
intervalBED(intervalSIB, intervalQuiet, distance1 = dminutes(30),
  distance2 = dminutes(5))
```

Arguments

intervalQuiet	dataframe of intervals where the activity is low enough to be considered as compatible with being in a bed
distance1	distance allowed in SIB intervals to consider that the form part of the same sleep period and not different sleeping periods
distance2	distance allowed in intervalQuiet to connect intervals of low activity, considered as taking part in the same low activity period.
intervalSib	dataframe of intervals considered as SIB

Value

A dataframe of intervals representing the intervals of being bed and sleeping for every day of accelerometry data

---

intervalIntersect *Intersect the intervals given by two dataframes Generates the intersection dataframe resulting from the intersection of the intervals represented by two dataframes of intervals*

Description

Intersect the intervals given by two dataframes Generates the intersection dataframe resulting from the intersection of the intervals represented by two dataframes of intervals

Usage

```
intervalIntersect(interval1, interval2)
```

Arguments

interval1	first dataframe
interval2	second dataframe

Value

A dataframe of intervals representing the inersection

Index