Package 'HumanActivityDetection'

May 12, 2020

Type Package

Title What the Package Does (Title Case)

Version 0.1.0

Author Denis Hock

Maintainer Denis Hock <dehock@fb2.fra-uas.de>

Description This package is part of the article
'Entropy-Based Metrics for Human Activity Recognition using Energy Demand'.
It contains data from (originally from the ECO dataset) with labelled human activity.
Aside from the entropy inspired metrics there five other methods to detect human activity.

License LGPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Depends R (>= 2.10)

R topics documented:

acc *Accuracy*

Description

(tp + dtn) / (tp + tn + fp + fn)

Usage

```
acc(data)
```

Arguments

data	confusionmatrix: data.frame(tp, tn, fp, fn)

Value

Accuracy [0,1]

Examples

```
actual <- c(0,0,1,1)
predict <- c(1,0,1,0)
result <- confusionmatrix(actual,predict)
acc(result)
```

changeLabelResolution *Change resolution of labelled Data.*

Description

Returns the labelled dataset in a different time resolution

Usage

```
changeLabelResolution(labelDataset, mins, aggMethod = max)
```

Arguments

labelDataset	the initial dataset
mins	the new resolution in minutes
aggMethod	the function to aggregate the power (W) values

Value

returns a data.frame with time, power and activity labels (0/1)

Examples

```
data(house1Summer)
changeLabelResolution(house1Summer,15)
```

combineApps *Merge ECO Appliances into a dataframe*

Description

According to the folder structure of the ECO dataset, this function interates through the directories of all appliances and reads each available csv-file (1 per day). all appliances are merged into a single data.frame (time | app1 | app2 | app3)

!!The Working directory must be set to the ECO dataset!!

We assume the following structure: e.g.: "Electricity Consumption and Occupancy/household 1/01_plugs_csv/01/02" for App2 in Household 1.

Usage

```
combineApps(House, NrOfApps)
```

Arguments

House	String with the number of the house (e.g. '01'- '06')
NrOfApps	the number of appliance-folders in this house

Value

returns a data.frame with time, and per appliance a column with power (W)

Examples

```
## Not run:
setwd("./Path/to/Electricity Consumption and Occupancy")
house2 <- combineApps('02',12)
save(house1, file=paste('h1','.Rda',sep=''))

## End(Not run)
```

confusionmatrix *Generate a Confusion Matrix (tp,tn,fp,fn)*

Description

Values to compute Accuracy

Usage

```
confusionmatrix(actual, predict)
```

Arguments

actual	labelled data
predict	result of the human activity detection

Value

returns a data.frame tp,tn,fp,fn

Examples

```
actual <- c(0,0,1,1)
predict <- c(1,0,1,0)
confusionmatrix(actual,predict)
```

---

geoMa *Geometric Moving Avg.*

Description

According to Kleiminger et. al.

Usage

```
geoMa(vector, delta = 0.0015)
```

Arguments

vector	energy demand
delta	numeric [0,1]

Value

detected activity (0/1)

Examples

```
data(house1Winter)
geoMa(changeLabelResolution(house1Winter,15)$power)
```

---

house1Summer *House 1, Summer, from 2012-06-01 to 2012-10-01, 175680 rows Power (W) measurements of the ECO dataset with human activity labels.*

Description

House 1, Summer, from 2012-06-01 to 2012-10-01, 175680 rows Power (W) measurements of the ECO dataset with human activity labels.

Usage

```
data(house1Summer)
```

Format

A data frame with 3 variables:

time posixCT timestamp

power Watt as numeric

activity human activity YES=1/NO=0 as numeric

Source

http://vs.inf.ethz.ch/res/show.html?what=eco-data

See Also

house1Winter, house2Summer, house2Winter ... , house6Winter

`house1Winter` *House 1, Winter, from 2012-10-01 to 2013-01-31, 175680 rows Power (W) measurements of the ECO dataset with human activity labels.*

Description

House 1, Winter, from 2012-10-01 to 2013-01-31, 175680 rows Power (W) measurements of the ECO dataset with human activity labels.

Usage

```
data(house1Summer)
```

Format

A data frame with 3 variables:

time posixCT timestamp

power Watt as numeric

activity human activity YES=1/NO=0 as numeric

Source

http://vs.inf.ethz.ch/res/show.html?what=eco-data

See Also

house1Winter, house2Summer, house2Winter ... , house6Winter

`intervalEntropy` *IntervalEntropy*

Description

Computes the Entropy vertically

Usage

```
intervalEntropy(
  vector,
  threshold = 0.1,
  windowSize = 15,
  cuts = 22,
  fun = "uniform"
)
```

Arguments

`vector`	energy demand
`threshold`	numeric [0,1]
`windowSize`	size of the time-window
`cuts`	number of intervals
`fun`	can be: uniform, cluster, log

Value

detected activity (0/1)

`label` *Label human activity using the appliance level data*

Description

The condition to consider an appliance is: C1 Median is closer to Min than to Max C2 Max is greater than 10 C3 Third quartil is closer to Median than Max If the appliance is considered, we select values above mean, greater 1. Note that the function should be used for a single day and not the complete dataset, as that may influence the mean-value

Usage

```
label(data)
```

Arguments

`data`	requires the output of function 'combineApps' ('PosixCT','numeric',...'numeric')

Value

returns a data.frame with time, power and activity labels (boolean)

Examples

```
## Not run:
setwd("./Path/to/Electricity Consumption and Occupancy")
house1 <- combineApps('01',7)
labelData <- label(house1)
allLabels <- apply(labelData[,2:length(labelData)],1,any)
result <- data.frame(house1[,1],labelData[,1],allLabels)
names(result) <- c('time','power','activity')
save(result, file='h1_label.Rda')

## End(Not run)
```

`movingAvg` *Moving Avg. and threshold*

Description

Moving Avg. higher or lower threshold is detected as activity

Usage

```
movingAvg(vector, threshold = NULL)
```

Arguments

`vector`	energy demand
`threshold`	is per default the standard deviation of 'vector'

Value

detected activity (0/1)

Examples

```
data(house1Summer)
resHouse1Summer <- changeLabelResolution(house1Summer,15,mean)
r1 <- pht(resHouse1Summer$power)
r2 <- geoMa(resHouse1Summer$power)
r3 <- movingAvg(resHouse1Summer$power)
r4 <- trendThreshold(resHouse1Summer$power)
r5 <- stdThreshold(house1Summer$power)
r6 <- intervalEntropy(house1Summer$power)
r7 <- slidingWindowEntropy(house1Summer$power)
a1 <-acc(confusionmatrix(resHouse1Summer$activity, r1))
a2 <-acc(confusionmatrix(resHouse1Summer$activity, r2))
a3 <-acc(confusionmatrix(resHouse1Summer$activity, r3))
a4 <-acc(confusionmatrix(resHouse1Summer$activity, r4))
a5 <-acc(confusionmatrix(resHouse1Summer$activity, r5))
a6 <-acc(confusionmatrix(resHouse1Summer$activity, r6))
```

```
a7 <-acc(confusionmatrix(resHouse1Summer$activity, r7))
result <- c(a1, a2, a3, a4, a5, a6, a7)
barplot(result)
```

---

pht *Page–Hinkley test.*

Description

According to Kleiminger et. al.

Usage

```
pht(vector, detectThreshold = 1000, magThreshold = 100)
```

Arguments

vector	energy demand
detectThreshold	numeric in Watt
magThreshold	numeric in Watt

Value

detected activity (0/1)

Examples

```
data(house1Summer)
pht(changeLabelResolution(house1Summer,15)$power)
```

---

slidingWindowEntropy *Sliding Window Entropy*

Description

Computes the Entropy horizontally

Usage

```
slidingWindowEntropy(vector, threshold = 0.001, windowSize = 15)
```

Arguments

vector	energy demand
threshold	numeric [0,1]
windowSize	size of the time-window

Value

detected activity (0/1)

stdThreshold *standard deviation and threshold*

Description

Detects activity if the std. of the window > threshold

Usage

```
stdThreshold(vector, threshold = 25, windowSize = 15)
```

Arguments

vector	energy demand
threshold	numeric in Watt
windowSize	size of the time-window

Value

detected activity (0/1)

trendThreshold *trendThreshold*

Description

uses the trend component of time series decomposition

Usage

```
trendThreshold(vector, threshold = NULL)
```

Arguments

vector	energy demand
threshold	is per default the mean of 'vector'

Value

detected activity (0/1)

Index