

Package ‘DTWUMI’

January 20, 2025

Type Package

Title Imputation of Multivariate Time Series Based on Dynamic Time Warping

Version 1.0

Date 2018-06-12

Author DEZECACHE Camille, PHAN Thi Thu Hong, POISSON-CAILLAULT Emilie

Maintainer POISSON-CAILLAULT Emilie <emilie.poisson@univ-littoral.fr>

Description

Functions to impute large gaps within multivariate time series based on Dynamic Time Warping methods. Gaps of size 1 or inferior to a defined threshold are filled using simple average and weighted moving average respectively. Larger gaps are filled using the methodology provided by Phan et al. (2017) <DOI:10.1109/MLSP.2017.8168165>: a query is built immediately before/after a gap and a moving window is used to find the most similar sequence to this query using Dynamic Time Warping. To lower the calculation time, similar sequences are pre-selected using global features. Contrary to the univariate method (package 'DTWBI'), these global features are not estimated over the sequence containing the gap(s), but a feature matrix is built to summarize general features of the whole multivariate signal. Once the most similar sequence to the query has been identified, the adjacent sequence to this window is used to fill the gap considered. This function can deal with multiple gaps over all the sequences composing the input multivariate signal. However, for better consistency, large gaps at the same location over all sequences should be avoided.

Depends R (>= 3.0.0)

Imports dtw, rlist, stats, e1071, entropy, lsa, DTWBI

License GPL (>= 2)

RoxygenNote 6.0.1

URL <http://mawenzi.univ-littoral.fr/DTWUMI/>

NeedsCompilation no

Repository CRAN

Date/Publication 2018-07-13 14:30:09 UTC

Contents

DTWUMI-package	2
dataDTWUMI	3
DTWUMI_1gap_imputation	4
DTWUMI_imputation	5
imp_1NA	6
Indexes_size_missing_multi	7
Index	8

DTWUMI-package	<i>Imputation of Multivariate Time Series Based on Dynamic Time Warping</i>
----------------	---

Description

Functions to impute large gaps within multivariate time series based on Dynamic Time Warping methods. Gaps of size 1 or inferior to a defined threshold are filled using simple average and weighted moving average respectively. Larger gaps are filled using the methodology provided by Phan et al. (2017) <DOI:10.1109/MLSP.2017.8168165>: a query is built immediately before/after a gap and a moving window is used to find the most similar sequence to this query using Dynamic Time Warping. To lower the calculation time, similar sequences are pre-selected using global features. Contrary to the univariate method (package 'DTWBI'), these global features are not estimated over the sequence containing the gap(s), but a feature matrix is built to summarize general features of the whole multivariate signal. Once the most similar sequence to the query has been identified, the adjacent sequence to this window is used to fill the gap considered. This function can deal with multiple gaps over all the sequences composing the input multivariate signal. However, for better consistency, large gaps at the same location over all sequences should be avoided.

Details

Index of help topics:

DTWUMI-package	Imputation of Multivariate Time Series Based on Dynamic Time Warping
DTWUMI_1gap_imputation	Imputation of a large gap based on DTW for multivariate signals
DTWUMI_imputation	Large gaps imputation based on DTW for multivariate signals
Indexes_size_missing_multi	Indexing gaps size
dataDTWUMI	A multivariate times series consisting of three signals as example for DTWUMI package
imp_1NA	Imputing gaps of size 1

Author(s)

DEZECACHE Camille, PHAN Thi Thu Hong, POISSON-CAILLAULT Emilie

Maintainer: POISSON-CAILLAULT Emilie <emilie.poisson@univ-littoral.fr>

References

Thi-Thu-Hong Phan, Emilie Poisson-Caillault, Alain Lefebvre, Andre Bigand. Dynamic time warping-based imputation for univariate time series data. Pattern Recognition Letters, Elsevier, 2017, <DOI:10.1016/j.patrec.2017.08.019>. <hal-01609256>

Examples

```
data(dataDTWUMI)
dataDTWUMI_gap <- dataDTWUMI[["incomplete_signal"]]
imputation <- DTWUMI_imputation(dataDTWUMI_gap, gap_size_threshold = 10, DTW_method = "DTW")
plot(dataDTWUMI_gap[, 1], type = "l", lwd = 2)
lines(imputation$output[, 1], col = "red")
plot(dataDTWUMI_gap[, 2], type = "l", lwd = 2)
lines(imputation$output[, 2], col = "red")
plot(dataDTWUMI_gap[, 3], type = "l", lwd = 2)
lines(imputation$output[, 3], col = "red")
```

dataDTWUMI

A multivariate times series consisting of three signals as example for DTWUMI package

Description

A multivariate times series consisting of three signals as example for DTWUMI package

Usage

```
dataDTWUMI
```

Format

A list storing two data frames with three columns each. The first table contains the original complete simulated data. The second table contains the same simulated data with one large gap added within each signal.

 DTWUMI_1gap_imputation

Imputation of a large gap based on DTW for multivariate signals

Description

Fills a gap of size 'gap_size' beginning at the position 'begin_gap' within a multivariate signal using DTW.

Usage

```
DTWUMI_1gap_imputation(data, id_sequence, begin_gap, gap_size,
  DTW_method = "DTW", threshold_cos = 0.995, thresh_cos_stop = 0.8,
  step_threshold = 2, ...)
```

Arguments

data	a multivariate signals containing gaps
id_sequence	id of the sequence containing the gap to fill (corresponding to the column number)
begin_gap	id of the beginning of the gap to fill
gap_size	size of the gap to fill
DTW_method	DTW method used for imputation ("DTW", "DDTW", "AFBDTW"). By default "DTW"
threshold_cos	threshold used to define similar sequences to the query
thresh_cos_stop	Define the lowest cosine threshold acceptable to find a similar window to the query
step_threshold	step used within the loops determining the threshold and the most similar sequence to the query
...	additional arguments from dtw() function

Value

returns a list containing the following elements:

- imputed_values: output vector containing the imputation proposal
- id_imputation: a vector containing the position of the imputed values extracted
- id_sim_win: a vector containing the position of the similar window to the query
- id_gap: a vector containing the position gap considered
- id_query: a vector containing the position of the query

Author(s)

DEZECACHE Camille, PHAN Thi Thu Hong, POISSON-CAILLAULT Emilie

Examples

```

data(dataDTWUMI)
dataDTWUMI_gap <- dataDTWUMI[["incomplete_signal"]]
t <- 207 ; T <- 40
imputation <- DTWUMI_1gap_imputation(dataDTWUMI_gap, id_sequence=1, t, T)
plot(dataDTWUMI_gap[, 1], type = "l", lwd = 2)
lines(y = imputation$imputed_values, x = imputation$id_gap, col = "red")
lines(y = dataDTWUMI_gap[imputation$id_query, 1], x = imputation$id_query, col = "green")
lines(y = dataDTWUMI_gap[imputation$id_sim_win, 1], x = imputation$id_sim_win, col = "blue")
lines(y = dataDTWUMI_gap[imputation$id_imputation, 1], x = imputation$id_imputation, col = "orange")

```

DTWUMI_imputation *Large gaps imputation based on DTW for multivariate signals*

Description

Fills all gaps within a multivariate signal. Gaps of size 1 are filled using the average values of nearest neighbours. Gaps of size >1 and <gap_size_threshold are filled using weighted moving average. Larger gaps are filled using DTW.

Usage

```

DTWUMI_imputation(data, gap_size_threshold, DTW_method = "DTW",
  threshold_cos = 0.995, thresh_cos_stop = 0.8, step_threshold = 2, ...)

```

Arguments

data	a multivariate signals containing gaps
gap_size_threshold	threshold above which dtw based imputation is computed. Below this threshold, a weighted moving average is calculated
DTW_method	DTW method used for imputation ("DTW", "DDTW", "AFBDTW"). By default "DTW"
threshold_cos	threshold used to define similar sequences to the query
thresh_cos_stop	Define the lowest cosine threshold acceptable to find a similar window to the query
step_threshold	step used within the loops determining the threshold and the most similar sequence to the query
...	additional arguments from dtw() function

Value

returns a list containing a dataframe of completed signals

Author(s)

DEZECACHE Camille, PHAN Thi Thu Hong, POISSON-CAILLAULT Emilie

Examples

```
data(dataDTWUMI)
dataDTWUMI_gap <- dataDTWUMI[["incomplete_signal"]]
imputation <- DTWUMI_imputation(dataDTWUMI_gap, gap_size_threshold = 10)
plot(dataDTWUMI_gap[, 1], type = "l", lwd = 2)
lines(imputation$output[, 1], col = "red")
plot(dataDTWUMI_gap[, 2], type = "l", lwd = 2)
lines(imputation$output[, 2], col = "red")
plot(dataDTWUMI_gap[, 3], type = "l", lwd = 2)
lines(imputation$output[, 3], col = "red")
```

imp_1NA

Imputing gaps of size 1

Description

Imputes isolated missing values based on the average of nearest neighbours.

Usage

```
imp_1NA(data, pos1)
```

Arguments

data	a univariate signal
pos1	the position of the beginning of gaps of size 1, as obtained using <code>Indexes_size_missing_multi()</code> function

Value

returns a new vector of same size with imputed values

Author(s)

DEZECACHE Camille, PHAN Thi Thu Hong, POISSON-CAILLAULT Emilie

Indexes_size_missing_multi
Indexing gaps size

Description

Stores the position of the beginning of each gap and their respective size within a multivariate signal.

Usage

```
Indexes_size_missing_multi(data)
```

Arguments

data multivariate signal

Value

returns a list with one element per signal. Within each element of this list, the first column gives the position of the beginning of each gap and the second column its size.

Author(s)

DEZECACHE Camille, PHAN Thi Thu Hong, POISSON-CAILLAULT Emilie

Examples

```
data(dataDTWUMI)  
id_NA <- Indexes_size_missing_multi(dataDTWUMI$incomplete_signal)
```

Index

- * **DTW**
 - DTWUMI-package, [2](#)
 - * **datasets**
 - dataDTWUMI, [3](#)
 - * **imputation**
 - DTWUMI-package, [2](#)
 - * **package**
 - DTWUMI-package, [2](#)
 - * **times series**
 - DTWUMI-package, [2](#)
- dataDTWUMI, [3](#)
DTWUMI (DTWUMI-package), [2](#)
DTWUMI-package, [2](#)
DTWUMI_1gap_imputation, [4](#)
DTWUMI_imputation, [5](#)
- imp_1NA, [6](#)
Indexes_size_missing_multi, [7](#)