newcolorrgb.8,.349,.1

Dynamic Time Warping-based imputation for univariate time series data

Thi-Thu-Hong PHAN[a,b,*], Émilie POISSON CAILLAULT[a,c,*], Alain LEFEBVRE[c], André BIGAND[a]

[a] *Univ. Littoral Côte d'Opale, EA 4491-LISIC, F-62228 Calais, France*
[b] *Vietnam National University of Agriculture, Department of Computer Science, Hanoi, Vietnam*
[c] *IFREMER, LER BL, F-62321 Boulogne-sur-mer, France*

Abstract

Time series with missing values occur in almost any domain of applied sciences. Ignoring missing values can lead to a loss of efficiency and unreliable results, especially for large missing sub-sequence(s). This paper proposes an approach to fill in large gap(s) within time series data under the assumption of effective information. To obtain the imputation of missing values, we find the most similar sub-sequence to the sub-sequence before (resp. after) the missing values, then complete the gap by the next (resp. previous) sub-sequence of the most similar one. Dynamic Time Warping algorithm is applied to compare sub-sequences, and combined with the shape-feature extraction algorithm for reducing insignificant solutions. Eight well-known and real-world data sets are used for evaluating the performance of the proposed approach in comparison with five other methods on different indicators. The obtained results proved that the performance of our approach is the most robust one in case of time series data having high auto-correlation and cross-correlation, strong seasonality, large gap(s), and complex distribution.

Keywords: Imputation, Missing data, Univariate time series, DTW, Similarity

1. Introduction

Recent advances in monitoring systems, communication and information technology, storage capacity and remote sensing systems make it possible to consider huge time series databases. These databases have been collected over many years with intraday samplings. However, they are usually incomplete due to sensor failures, communication/transmission problems or bad weather conditions for manual measures or maintenance. This is particularly the case for marine samples (Rousseeuw et al. (2013), Ceong et al. (2012)). Incomplete missing data are problematic (Gómez-Carracedo et al. (2014)) because most data analysis algorithms and most statistical softwares are not designed to handle this kind of data.

Let consider some terminologies and a real marine data set to illustrate the problem. A time series $x = \{x_t | t = 1, 2, \cdots, N\}$ is a set of N observations successive indexed in time, occurring in uniform intervals. A single hole at index t is an isolated missing value where observations at time $t-1$ and $t+1$ are available, we note $x_t = NA$ (NA stands for not available). A hole of size T, also called gap, is an interval $[t : t+T-1]$ of consecutive missing values and is denoted $x[t : t+T-1] = NA$. We define a large gap when T is larger than the known-process change, so it depends on each application. At the MAREL Carnot station, a marine water monitoring platform in the eastern English Channel, France (Lefebvre (2015)), 19 large time series are collected every 20 minutes as fluorescence, turbidity, oxygen saturation and so on. These data contain single and large holes. For example, oxygen saturation series has 131,472 observations and only 81.9% available. This series comprises 4,004 isolated missing values and many consecutive missing data. The size of these gaps are

*Corresponding authors:
Email addresses: ptthong@vnua.edu.vn (Thi-Thu-Hong PHAN), emilie.poisson@univ-littoral.fr (Émilie POISSON CAILLAULT)

various from one hour to few months; the largest gap is a 3,044 points corresponding to 42 days. Single holes and gaps having $T <$ tide duration-holes (807 missing points) could be easily replaced by local averages. For the other gaps, the phytoplankton bloom dynamics or composition changes too fast to use linear or spline imputation method.

Other classical solution consists in ignoring missing data or listwise deletion. But it is easy to imagine that this drastic solution may lead to serious problems, especially for time series data (the considered values would depend on the past values). The first potential consequence of this method is information loss which could lose efficiency (Noor et al. (2014)). The second consequence is about systematic differences between observed and unobserved data that leads to biased and unreliable results (Hawthorne and Elliott (2005)).

Therefore, it is crucial to propose a new technique to estimate missing values. One prospective approach to solve missing data problems is the adoption of imputation techniques (Junninen et al. (2004)). These techniques should ensure that the obtained results are efficient (having minimal standard errors) and reliable (effective, curve-shape respect).

According to our knowledge, there is no application for filling time series data with large missing gap(s) size for univariate time series. We therefore investigate and propose an algorithm to complete large gap(s) of univariate time series based on Dynamic Time Wrapping (Sakoe and Chiba (1978)). We do not deal with all the missing data over the entire series, but we focus on each large gap where series-shape change could occur over the duration of this large gap. Further, the distribution of missing values or entire signal could be very difficult to estimate, so it is necessary to make some assumptions. Our approach makes the assumption that the information about missing values exists within the univariate time series and takes into account the time series characteristics.

This paper is organized as follows. First,we discuss the related work in section 2. The analysis of time series data is discussed in Section 3. The proposed approach is introduced in Section 4. Experimental results and discussion on 8 data sets are illustrated in Section 5. Conclusion is set out in Section 6.

2. Related work

In the literature, missing data mechanisms can be divided into three categories. Each category is based on one possible cause: "Missing data are completely random" (Missing Completely At Random, MCAR, in the literature), "Missing data are random" (Missing At Random, MAR) and "Missing data are not random" (Not Missing At Random, NMAR) (Little and Rubin (2014)). It is important to understand the causes that produce missing data to develop an imputation task. This can help to select an appropriate imputation algorithm (Moritz et al. (2015)). But in practice, understanding the causes remains a challenging task when missing data cannot be known at all, or when these data have a complex distribution (Gómez-Carracedo et al. (2014)). Similarly, assigning sub-sequences of missing values to a category can be blurry (Moritz et al. (2015)). Commonly, most current research works focus on the three types of missing data previously defined to find out corresponding imputation methods. Regarding imputation methods, a large number of successful approaches have been proposed for completing missing data.

Concerning the imputation task for multivariate time series, many studies have been investigated using machine learning techniques as Shah et al. (2014), Liao et al. (2014), Rahman et al. (2015) and model techniques such as Raghunathan and Siscovick (1996), Schafer (1997), Van Buuren et al. (1999), Raghunathan et al. (2001), Royston (2007), Joseph et al. (2009), Stuart et al. (2009), Lee and Carlin (2010), Spratt et al. (2010), Gelman et al. (2015), Deng et al. (2016). The efficiency of these algorithms is based on correlations between signals or their features, and missing values are estimated from the observed values. However, handling missing values within univariate time series data differs from multivariate time series techniques. We must only rely on the available values of this unique variable to estimate the incomplete values of the time series. Moritz et al. (2015) showed that imputing univariate time series data is a particularly challenging task.

Fewer studies are devoted to the imputation task for univariate time series. Allison (2001) and Bishop (2006) proposed to simply substitute the mean or the median of available values to each missing value. These simple algorithms provide the same result for all missing values

leading to bias result and to undervalue standard error (Crawford et al. (1995), Sterne et al. (2009)). Other imputation techniques for univariate time series are linear interpolation, spline interpolation and the nearest neighbor interpolation. These techniques were studied for missing data imputation in air quality data sets (Junninen et al. (2004)). The results showed that univariate methods are dependent upon the size of the gap in time: the larger gap, the less effective technique. Walter et al. (Walter.O et al. (2013)) carried out a performance comparison of three methods for univariate time series, namely, ARIMA (Autoregressive Integrated Moving Average), SARIMA (Seasonal ARIMA), and linear regression. The linear regression method was more efficient and effective than the other two methods, only when rearranging the data in periods. This study treated non-stationary seasonal time series data but it did not take into account series without seasonality. Chiewchanwattana et al. proposed the Varied-Window Similarity Measure (VWSM) algorithm (Chiewchanwattana et al. (2007)). This method is better than the spline interpolation, the multiple imputation, and the optimal completion strategy fuzzy c-means algorithms. However, this research only focused on filling one isolated missing value, but did not consider sub-sequence missing. Moritz et al. (2015) performed an overview about univariate time series imputation comparing six imputation methods. Nevertheless, this study only considered the MCAR type.

3. Time series characterization

Filling large gaps within time series requires firstly to characterize the data. This step permits to extract useful information from the data set and makes the data set easily exploitable. The four specific components of time series are trend, seasonal, cyclical and random change:

1. *Trend component*: That is the change of variable(s) in terms of monitoring for a long time. If there exists a trend within the time series data (i.e. on the average data), the measurements tend to increase (or decrease) over time.
2. *Seasonal component*: This component takes into account intra-interval fluctuations. That means there is a regular and repeated pattern of peaks and valleys within the time series related to a calendar period such as seasons, quarters, months, weekdays, and so on.
3. *Cyclical component*: This component equals the seasonal one, the difference is that its cycle duration is more than one year.
4. *Random change component*: This component considers random fluctuations around the trend; this could affect the cyclical and seasonal variations of the observed sequence, but it cannot be predicted by previous data (in the past of time series).

There are different techniques to decompose time series into components. "Decompose a time series into seasonal, trend and irregular components using moving averages" (R-starts package, R Core Team (2016)) is the most common technique. In this study, we use this technique to analyze time series data.

Auto-correlation function (ACF) provides an additional important indication of the properties of time series (i.e. how past and future data points are related). Therefore, it can be used to identify the possible structure of time series data, and to create reliable forecasts and imputations (Moritz et al. (2015)). High auto-correlation values mean that the future is strongly correlated to the past. Fig. 1 indicates the auto-correlation of Mackey-Glass chaotic, water level and Google data sets in our experiment.

4. The proposed method - DTWBI

In this part, we present a new method for imputing missing values of univariate time series data.

A time series x is referred as incomplete time series when it contains missing values (or values are Not Available-NA). Recall that the portion of a time series between two points x_t and x_{t+T-1} with $x_i = NA$ ($i = t : t + T - 1$) is called a gap of T-size at position t. In this paper, we consider a large gap when $T \geq 6\%N$ for small time series ($N < 10,000$) or when T is larger than the known-process change.

The proposed approach finds the most similar subsequence (Qs) to a query (Q), with Q (cf. Fig. 2) is the sub-sequence before a gap of T size at position t ($Q = x[t - T : t - 1]$), and completes this gap by the following sub-sequence of the Qs.

To find the Qs similar sub-sequence, we use the principles of Dynamic Time Warping - DTW (Sakoe and Chiba

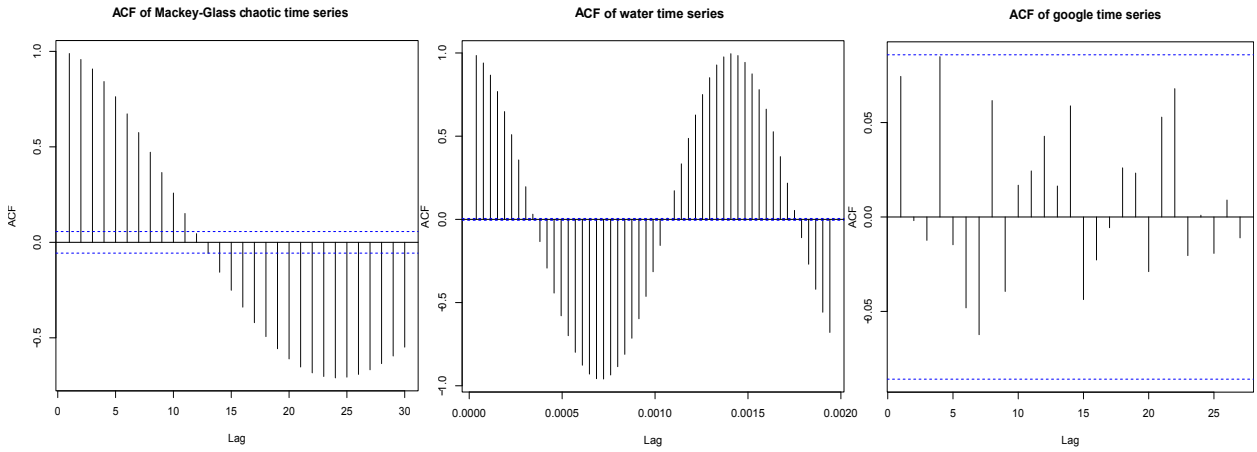

Figure 1: ACF of Mackey-Glass chaotic, water level and Google time series

(1978)), especially transformed from original data to Derivative Dynamic Time Warping - DDTW data (Keogh and Pazzani (2001)). The DDTW data are used because we can obtain information about the shape of sequence (Keogh and Pazzani (2001)). The dynamics and the shape of data before a gap are a key-point of our method. The elastic matching is used to find a similar window to the Q query of T size in the search database. Once the most similar window is identified, the following window will be copied to the location of missing values. Fig. 2 describes the different steps of our approach.

The detail of DTWBI (namely DTW-Based Imputation) algorithm is introduced in Algorithm 1. In the proposed method, the shape-feature extraction algorithm (Phan et al. (2016)) is applied before using DTW algorithm in order to reduce the computation time. As we know DTW time complexity is $O(N^2)$, so this is a very useful step to decrease computation time of DTW method. A reference window is selected to calculate DTW cost only if the correlation between the shape-features (also called the global features) of this window and the ones of the query is very high. In addition, we apply the shape-feature extraction algorithm because it better presents the shape and dynamics of series through 9 elements, such as moments (the 1^{st} moment, the 2^{nd} moment, the 3^{rd} moment), number of peaks, entropy, etc (see Phan et al. (2016) for more detail). This is an important objective of the proposed method. In Algorithm 1, we just mention the finding of similar windows before the gap. In case of finding similar windows after the gap, the method just needs to shift the corresponding index.

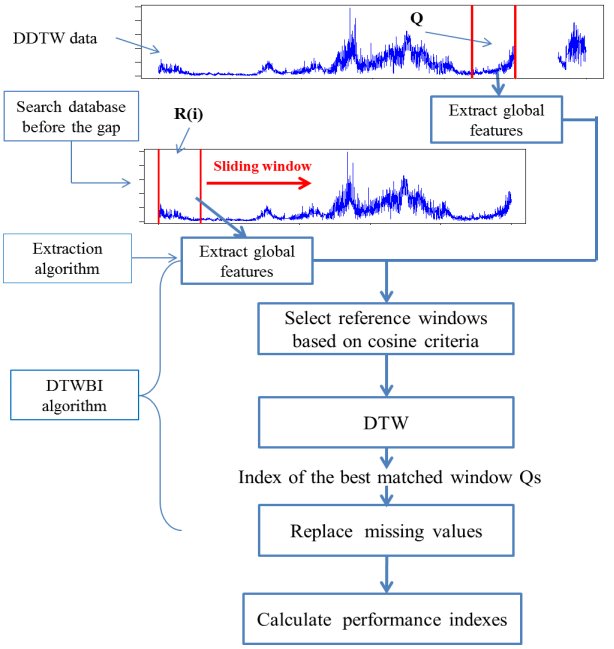

Figure 2: Diagram of DTWBI method for univariate time series imputation

5. Experimental results and discussion

5.1. Data presentation

In this study, we analyzed 8 data sets in order to evaluate the performance of the proposed technique. 4 data sets come from TSA package (Hyndman and Khandakar (2008)). These data sets are chosen because they are usually used in the literature, including Airpassenger, Beersales, Google, and SP. Besides, we also choose other data sets from various domains in different places:

1. Airpassenger - Monthly total international airline passengers from 01/1960 to 12/1971.
2. Beersales - Monthly beer sales in millions of barrels, from 01/1975 to 12/1990.
3. Google - Daily returns of the google stock from 08/20/04 to 09/13/06.
4. SP - Quarterly S&P Composite Index, 1936Q1 - 1977Q4.
5. CO2 concentrations - This data set contains monthly mean CO2 concentrations at the Mauna Loa Observatory from 1974 to 1987 (Thoning et al. (1989)).
6. Mackey-Glass chaotic - The data is generated from the Mackey-Glass equation which is the nonlinear time delay differential (Mackey and Glass (1977)).
7. Phu Lien temperature - This data set is composed of monthly mean air temperature at the Phu Lien meteorological station in Vietnam from 1/1961 to 12/2014.
8. Water level - The MAREL Carnot data in France acquired from 2005 up today. For our study, we focus on the water level, sampling frequency of 20 minutes from 01/1/2015 to 31/12/2009 (Lefebvre (2015)).

Table 1 summarizes characteristics of the data sets.

Table 1: Data characteristics

N0	Data set name	N0 of instants	Trend (Y/N)	Seasonal (Y/N)	Frequency
1	Air passenger	144	Y	Y	Monthly
2	Beersales	192	Y	Y	Monthly
3	Google	521	N	N	Daily
4	SP	168	Y	Y	Quarterly
5	CO2 concentrations	160	Y	Y	Monthly
6	Mackey-Glass chaotic	1201	N	N	
7	Phu Lien temperature	648	N	Y	Monthly
8	Water level	131472	N	Y	20 minutes

5.2. Univariate time series imputation algorithms

The performance of the proposed method compared with 5 other existing methods for univariate time series (namely, na.interp, na.locf, na.approx, na.aggregate, na.spline) is evaluated in this paper. All these methods are implemented using R language (na stands for Not Available):

1. na.interp (forecast R-package): linear interpolation for non-seasonal series and Seasonal Trend decomposition using Loess (STL decomposition) for seasonal series to replace missing values (Hyndman and Khandakar (2008)). A seasonal model is fit to the data, and then interpolation is made on the seasonally adjusted series, before re-seasonalizing. So, this method is especially devoted to strong and clear seasonality data.
2. na.locf (last observation carried forward) (zoo R-package): any missing value is replaced by the most recent non-NA value prior to it (Zeileis and Grothendieck (2005)). Conceptually, this method assumes that the outcome would not change after the last observed value. Therefore, there has been no time effect since the last observed data.
3. na.approx (zoo R-package): generic function for replacing each NA with interpolated values (Zeileis and Grothendieck (2005)).
4. na.aggregate (zoo R-package): generic function for replacing each NA with aggregated values. This allows imputing using the overall mean, by monthly means, etc (Zeileis and Grothendieck (2005)). In our experiment, we use the overall mean.
5. na.spline (zoo R-package): polynomial (cubic) interpolation to fill in missing data (Zeileis and Grothendieck (2005)).

5.3. Imputation performance indicators

After the completion of missing values, we assess the performance of our method, and then compare it with existing imputation methods based on four different metrics described as follows:

1. Similarity: $Sim(y, x)$ indicates the similarity between actual data (X) and imputation data (Y). It is calculated by:

$$Sim(y, x) = \frac{1}{T} \sum_{i=1}^{T} \frac{1}{1 + \frac{|y_i - x_i|}{max(x) - min(x)}} \qquad (1)$$

Where T is the number of missing values. A higher similarity (similarity value $\in [0, 1]$) highlights a better ability method for the task of completing missing values.

2. NMAE: The Normalized Mean Absolute Error between the imputed value y and the respective true value time series x is computed as:

$$NMAE(y, x) = \frac{1}{T} \sum_{i=1}^{T} \frac{|y_i - x_i|}{V_{max} - V_{min}} \quad (2)$$

Where V_{max}, V_{min} are the maximum and the minimum values of input time series (time series has missing data) by ignoring the missing values. A lower NMAE means better performance method for the imputation task.

3. RMSE: The Root Mean Square Error is defined as the average squared difference between the imputed value y and the respective true value time series x. This indicator is very useful for measuring overall precision or accuracy. In general, the most effective method would have the lowest RMSE.

$$RMSE(y, x) = \sqrt{\frac{1}{T} \sum_{i=1}^{T} (y_i - x_i)^2} \quad (3)$$

4. FSD: Fraction of Standard Deviation of the imputed value y and the respective true value time series x is defined as follows:

$$FSD(y, x) = 2 * \frac{|SD(y) - SD(x)|}{SD(y) + SD(x)} \quad (4)$$

This fraction indicates whether a method is acceptable or not (here SD stands for Standard Deviation). For the imputation task, FSD should be closer to 0, the imputation values are closer to the real values.

5.4. Experiment protocol

Indeed, we could not compare the ability of imputation algorithms on real missing data because the true values are not available. Therefore, we have to create simulated missing gaps on full data to compare the performance of imputation algorithms. For assessing the results, we use a technique based on three steps. In the first step, we create artificial missing data by deleting data values from known time series. The second step consists in applying the imputation algorithms to complete missing data. Finally, the third step compares the performance of the proposed method with published methods using the different imputation performance indicators as previously defined.

In the present study, 5 missing data levels are considered on 8 data sets. If the size of a data set (number of instants of the data set) is less than or equal to 10,000 samples, we create gaps with different sizes: 6%, 7.5%, 10%, 12.5%, 15% of overall data set size. In contrast, when the size of a data set is greater than 10,000 sampling points, gaps are built at rates 0.6%, 0.75%, 1%, 1.25%, and 1.5% of the data set size (here the largest gap of the water level time series is 1,972 missing values, corresponding to the missing rate 1.5%). For each missing rate, the algorithms are conducted 10 times by randomly selecting the missing positions on the data. We then run 50 iterations for each data set.

5.5. Results and discussion

5.5.1. Comparison of quantitative performance

Table 2 shows imputation average results of DTWBI, na.interp, na.locf, na.approx, na.aggregate, na.spline methods applied on 8 data sets using 4 indicators: similarity, NAME, RMSE, FSD.

- **Airpassenger, Beersales, Google, SP data sets**

The Airpassenger data set has both trend and seasonality components. The result from Table 2 indicates that when the gap size is greater than or equal to 10%, the proposed method has the highest similarity and the lowest NMAE and RMSE.

On the Beersales data set, considering similarity and RMSE indicators: na.interp method provides the best result and the second one is our approach. By contrast to these two indicators, our method has better results on NMEA and FSD indicators at any missing rate. When comparing na.interp method to the na.approx one on the Airpassenger and Beersales data sets, we can see na.interp shows better performance than na.approx method on any indicators and at every level of missing data. It corresponds to the fact that these two data sets have a clear seasonality component. Na.interp method takes into account the seasonality factor, so it can better handle seasonality

than na.approx does, although both algorithms use the interpolation for completing missing data.

On Airpassenger and Beersales data sets, na.aggregate approach gives less efficient results than na.interp. But on Google series, na.aggregate method yields the best performance: the highest similarity and the smallest NMEA, RMSE indicators. Without any trend on this data set, this method leads to the best result. For SP data set, na.aggegate method still highlights a good performance on NMEA and RMSE, but this approach has lower similarity than it has on Google series. The na.aggegate method replaces missing values by overall mean. However, SP series has a clear trend; therefore, na.aggregate method seems not to be effective with series having a strong trend.

In all data sets, FSD value of na.aggregate and na.locf methods always equals 2, because they use the same value for all missing data (last value for na.locf method; overall mean for na.aggregate).

- **CO2 concentrations, Mackey-Glass chaotic, Phu Lien temperature, water level data sets**

These data sets have a seasonality component (except Mackey-Glass chaotic series but this data set is regularly repeated), without any trend (excluding CO2 concentrations data set) and high auto-correlation. Our method demonstrates the best ability for completing missing data on these series: the highest similarity, the lowest NMAE, RMSE and FSD at any missing level. Furthermore, on Airpassenger, Beersales, Google and SP data sets, the similarity of our approach is lower, but the difference value in this indicator between the proposed method and the best method is small. On the contrary, for these four data sets, our method outperforms the existing techniques on any indicator and at any missing rate. The different values of these indicators between the proposed method and the other ones are quite large. The results confirm that the imputation values generated from the proposed method are close to the real values on data sets having high auto-correlation (see Fig. 1, the ACF maximum values of water and chaotic series are approximate 1), which means that there is a strong relationship between the available and the unknown values. Following the proposed method, the second one is na.aggregate one applied on the Mackey-Glass chaotic series, Phu Lien temperature and water level series. As mentioned above (Table 1), these data sets have no trend, that is why na.aggregate could demonstrate its ability. However, on the C02 series with clear trend, fully opposed to these 3 data sets, the performance of this method is the worst one.

Although na.interp method is well indicated for handling data sets with seasonality component: here with these 4 data sets this approach does not illustrate its capability. It gives the same results as na.approx method and lower results than our approach and the na.aggregate one (on the Mackey-Glass chaotic, Phu Lien temperature and water series). For any data set, na.spline method indicates the lowest performance. However on the water series, this method has the least performance for completing missing values. This means that the spline method is not suitable for this task.

5.5.2. Comparison of the visual performance

Table 2 indicates the quantitative comparison of 6 different methods for the task of completing missing values. In this part, Fig. 3, 4, 5, 7, and 8 show the comparison of visual imputation performance of different methods.

Fig. 3 presents the shape of imputation values of 5 existing methods (na.interp, na.locf, na.approx, na.aggregate and na.spline) with the true values at position 106, the gap size of 9 on the Airpassenger series. As we can notice on Table 2, considering low rates of missing data, the proposed approach is less effective than na.interp and na.aggregate methods for Airpassenger time series. However, when looking at Fig. 4, we find that the shape of the imputation values generated from DTWBI method is very similar to the shape of true values. Despite high similarity, low RMSE and NMAE, the shape of imputation values yielded from na.aggregate method (Fig. 3) is not as effective as the proposed method (Fig. 4). As analyzed above, the na.interp method better deals with seasonal factor, so their imputed values are asymptotic to the real values (Fig. 3).

Fig. 5 illustrates the visual comparison of DTWBI imputation values and real values on water level series at position 23,282, and at 0.6% rate of missing values (corresponding to 789 missing points). The proposed method proves again its capability for the task of completing missing values. We see that the shape of the imputation values generated from our method and the one of the true values

Table 2: Average imputation performance indexes of six methods on eight data sets

Gap size	Method	Airpassenger				Beersales				Google				SP			
		Sim	NMAE	RMSE	FSD	Sim	NMAE	RMSE	FSD	Sim	NMAE	RMSE	FSD	Sim	NMAE	RMSE	FSD
6%	DTWBI	0.777	0.034	21.1	0.24	0.88	**0.035**	0.7	**0.14**	0.83	0.14	0.034	**0.43**	0.74	0.026	35.5	0.7
	na.interp	**0.85**	**0.019**	**11.1**	**0.24**	**0.89**	0.063	**0.6**	0.15	0.83	0.11	0.032	1.11	0.74	0.028	36.3	0.54
	na.locf	0.76	0.044	26.3	2	0.81	0.129	1.2	2	0.81	0.126	0.036	2	0.75	0.022	29.2	2
	na.approx	0.77	0.037	21.8	1.01	0.8	0.136	1.3	1.5	0.83	0.11	0.032	1.11	0.73	0.028	37	1.03
	na.aggregate	0.8	0.033	20.1	2	0.83	0.11	1.1	2	**0.86**	**0.082**	**0.024**	2	**0.78**	**0.021**	**26.5**	2
	na.spline	0.71	0.057	35.1	0.52	0.68	0.26	2.3	0.55	0.5	1.813	0.473	1.02	0.63	0.045	56.8	**0.41**
7.5%	DTWBI	0.782	0.035	20.6	0.3	0.87	**0.038**	0.7	**0.1629**	0.84	0.131	0.032	**0.33**	0.76	0.03	38.9	**0.52**
	na.interp	**0.86**	**0.023**	**13.6**	**0.3**	**0.885**	0.067	**0.6**	0.163	0.83	0.119	0.034	1.18	0.78	0.024	33.1	0.67
	na.locf	0.77	0.046	27.4	2	0.81	0.123	1.2	2	0.82	0.126	0.035	2	0.77	0.026	34.8	2
	na.approx	0.74	0.053	31.3	1.49	0.8	0.132	1.3	1.51	0.83	0.119	0.034	1.18	0.78	0.025	34	1.1
	na.aggregate	0.81	0.033	20.2	2	0.82	0.112	1.1	2	**0.87**	**0.081**	**0.024**	2	**0.8**	**0.022**	**29.1**	2
	na.spline	0.6	0.112	65.4	0.45	0.6	0.404	3.5	0.43	0.44	3.652	0.963	1.38	0.69	0.042	54.5	0.55
10%	DTWBI	**0.887**	**0.02**	**12.7**	0.36	0.84	**0.054**	1	**0.13**	0.84	0.132	0.032	**0.23**	0.81	0.029	40.1	0.57
	na.interp	0.86	0.021	13.1	**0.34**	**0.89**	0.068	**0.7**	0.18	0.85	0.105	0.03	1.22	0.82	0.025	36.3	0.56
	na.locf	0.79	0.042	26.1	2	0.82	0.13	1.3	2	0.83	0.131	0.035	2	0.81	0.026	36.9	2
	na.approx	0.79	0.041	24.6	1.03	0.82	0.124	1.2	1.24	0.85	0.105	0.03	1.22	**0.83**	0.024	33.5	1.14
	na.aggregate	0.81	0.035	22.1	2	0.84	0.111	1.1	2	**0.87**	**0.084**	**0.024**	2	0.82	**0.023**	**31.7**	2
	na.spline	0.62	0.134	78.3	0.52	0.55	0.558	4.9	0.67	0.42	4.684	1.118	1.13	0.76	0.049	63.2	**0.45**
12.5%	DTWBI	**0.893**	**0.02**	**12.6**	**0.36**	0.87	**0.039**	0.7	**0.12**	0.85	0.138	0.032	**0.23**	0.8	0.03	41.9	0.61
	na.interp	0.86	0.023	14.8	0.39	**0.89**	0.068	**0.6**	0.15	0.85	0.115	0.032	1.27	0.81	0.028	38.8	**0.52**
	na.locf	0.8	0.044	26.9	2	0.82	0.127	1.2	2	0.84	0.129	0.035	2	0.81	0.027	36.1	2
	na.approx	0.79	0.043	26.7	0.95	0.8	0.147	1.4	1.28	0.85	0.115	0.032	1.27	**0.825**	0.027	35.6	1.06
	na.aggregate	0.82	0.035	21.8	2	0.84	0.109	1.1	2	**0.88**	**0.083**	**0.024**	2	0.824	**0.024**	**31**	2
	na.spline	0.64	0.129	76.8	0.67	0.61	0.458	4	0.77	0.39	2.143	0.532	1.4	0.61	0.113	132.4	0.69
15%	DTWBI	**0.895**	**0.02**	**12.8**	0.36	0.84	**0.054**	1	**0.1**	0.85	0.133	0.031	**0.29**	0.81	0.029	40.7	0.59
	na.interp	0.86	0.025	15.6	**0.35**	**0.89**	0.069	**0.7**	0.17	0.86	0.11	0.031	0.99	0.79	0.033	43.6	**0.49**
	na.locf	0.79	0.047	28.2	2	0.82	0.126	1.2	2	0.84	0.127	0.034	2	0.81	0.028	36.3	2
	na.approx	0.8	0.043	26.5	1.17	0.83	0.117	1.1	1.42	0.86	0.11	0.031	0.99	0.81	0.032	41	1
	na.aggregate	0.83	0.035	22.1	2	0.84	0.11	1.1	2	**0.89**	**0.079**	**0.023**	2	**0.82**	**0.025**	**32**	2
	na.spline	0.55	0.175	106.1	0.95	0.49	0.731	6.3	0.88	0.34	12.339	2.928	1.6	0.61	0.136	162.5	0.68
		CO2 concentrations				Mackey-Glass Chaotic				Phu Lien temperature				Water level			
6%	DTWBI	**0.93**	**0.001**	**0.3**	**0.04**	**0.95**	**0.005**	**0.01**	**0.03**	**0.88**	**0.06**	**1.7**	**0.08**	**0.95**	**0.009**	**0.1**	**0.05**
	na.interp	0.75	0.055	1.6	1.5	0.79	0.031	0.04	0.81	0.8	0.142	3.1	0.63	0.81	0.042	0.5	1.05
	na.locf	0.73	0.059	1.7	2	0.77	0.036	0.05	2	0.77	0.173	3.8	2	0.8	0.043	0.4	2
	na.approx	0.75	0.055	1.6	1.5	0.79	0.031	0.04	0.81	0.8	0.142	3.1	0.63	0.81	0.042	0.5	1.05
	na.aggregate	0.45	0.185	4.7	2	0.82	0.025	0.03	2	0.83	0.114	2.4	2	0.83	0.035	0.4	2
	na.spline	0.75	0.057	1.6	0.75	0.65	0.072	0.09	0.38	0.61	0.413	8.5	0.52	0.3	0.654	6.6	1.61
7.5%	DTWBI	**0.93**	**0.001**	**0.4**	**0.05**	**0.93**	**0.008**	**0.01**	**0.02**	**0.8788**	**0.061**	**1.7**	**0.06**	**0.96**	**0.007**	**0.1**	**0.02**
	na.interp	0.74	0.057	1.6	1.38	0.8	0.031	0.04	1.04	0.79	0.147	3.2	0.98	0.82	0.038	0.4	0.97
	na.locf	0.76	0.053	1.6	2	0.77	0.038	0.05	2	0.77	0.171	3.7	2	0.81	0.043	0.5	2
	na.approx	0.74	0.057	1.6	1.38	0.8	0.031	0.04	1.04	0.79	0.147	3.2	0.98	0.82	0.038	0.4	0.97
	na.aggregate	0.45	0.186	4.7	2	0.83	0.025	0.03	2	0.83	0.113	2.4	2	0.83	0.036	0.4	2
	na.spline	0.74	0.058	1.6	0.79	0.69	0.062	0.08	0.39	0.58	0.701	14.5	0.8	0.2	1.228	12	1.71
10%	DTWBI	**0.93**	**0.001**	**0.4**	**0.04**	**0.93**	**0.008**	**0.01**	**0.01**	**0.8791**	**0.063**	**1.8**	**0.05**	**0.97**	**0.005**	**0.1**	**0.03**
	na.interp	0.76	0.051	1.4	0.88	0.81	0.03	0.04	0.98	0.81	0.137	3	0.58	0.81	0.041	0.4	0.91
	na.locf	0.76	0.054	1.6	2	0.79	0.036	0.05	2	0.77	0.176	3.8	2	0.81	0.043	0.5	2
	na.approx	0.76	0.051	1.4	0.88	0.81	0.03	0.04	0.98	0.81	0.137	3	0.58	0.81	0.041	0.4	0.91
	na.aggregate	0.44	0.197	4.9	2	0.83	0.025	0.03	2	0.83	0.114	2.4	2	0.83	0.036	0.4	2
	na.spline	0.66	0.098	2.9	0.26	0.71	0.058	0.08	0.33	0.49	0.88	17.8	1.04	0.18	1.57	15.5	1.79
12.5%	DTWBI	**0.94**	**0.001**	**0.3**	**0.04**	**0.92**	**0.009**	**0.02**	**0.01**	**0.881**	**0.065**	**1.8**	**0.04**	**0.96**	**0.006**	**0.1**	**0.03**
	na.interp	0.78	0.049	1.5	1.39	0.8	0.033	0.04	1.13	0.79	0.163	3.5	1.44	0.81	0.044	0.5	1.21
	na.locf	0.75	0.057	1.7	2	0.79	0.036	0.05	2	0.78	0.18	3.8	2	0.81	0.043	0.5	2
	na.approx	0.78	0.049	1.5	1.39	0.8	0.033	0.04	1.13	0.79	0.163	3.5	1.44	0.81	0.044	0.5	1.21
	na.aggregate	0.44	0.2	5	2	0.84	0.025	0.03	2	0.84	0.116	2.4	2	0.83	0.036	0.4	2
	na.spline	0.71	0.073	2.2	0.38	0.61	0.093	0.12	0.63	0.55	0.653	13.7	0.99	0.25	0.96	9.8	1.74
15%	DTWBI	**0.94**	**0.001**	**0.3**	**0.04**	**0.92**	**0.01**	**0.02**	**0.01**	**0.882**	**0.066**	**1.8**	**0.05**	**0.96**	**0.007**	**0.1**	**0.04**
	na.interp	0.76	0.053	1.6	1.46	0.81	0.03	0.04	0.99	0.81	0.145	3.2	1	0.81	0.044	0.5	1.6
	na.locf	0.77	0.052	1.6	2	0.79	0.037	0.05	2	0.79	0.175	3.8	2	0.81	0.043	0.5	2
	na.approx	0.76	0.053	1.6	1.46	0.81	0.03	0.04	0.99	0.81	0.145	3.2	1	0.81	0.044	0.5	1.6
	na.aggregate	0.43	0.202	5.1	2	0.84	0.025	0.03	2	0.84	0.117	2.5	2	0.83	0.036	0.4	2
	na.spline	0.69	0.085	2.5	0.58	0.57	0.129	0.16	0.73	0.44	1.268	26.3	1.27	0.21	1.185	11.8	1.83

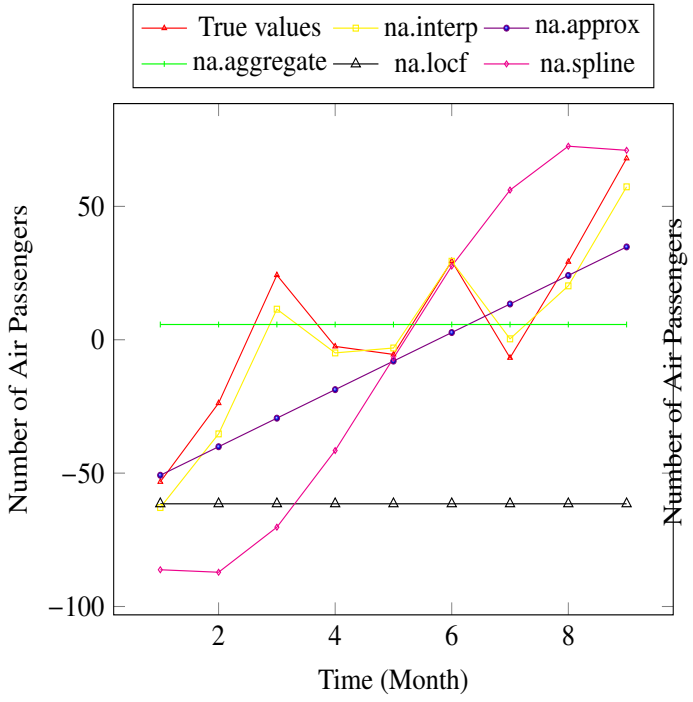

Figure 3: Visual comparison of imputed values of different imputation methods with true values on Airpassenger series at position 106 with the gap size of 9.

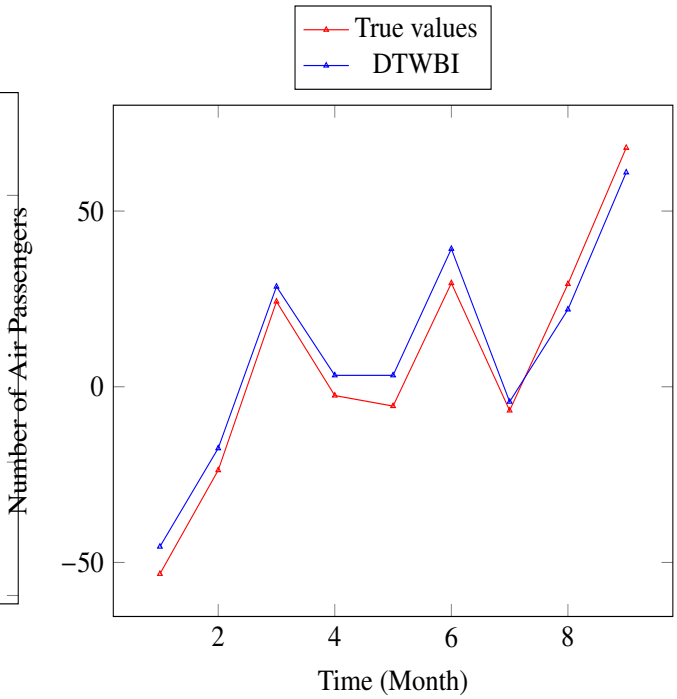

Figure 4: Visual comparison of imputed values of proposed method with true values on Airpassenger series at position 106 with the gap size of 9.

are almost completely identical. Fig. 6 shows the matching pairs between the query and the most similar reference window for the considered case. The values of matching pairs are very close, which indicates the reason why the DTWBI imputation values are very similar to the real values. In contrast to our approach, handling seasonal factor of na.interp method is ineffective on water level data set. This method does not provide good result such as on Airpassenger series (Fig. 3); its performance is the same as na.approx method (Fig. 7). Fig. 8 especially points out the obvious inefficiency of na.spline method for the task of completing missing values, considering series with high auto-correlation and large gap size (789 missing values in this case).

In this paper, we also calculate Cross-Correlation (CC) coefficients between the query with each reference window, and then we find the maximum coefficient. CC demonstrates that a pattern (here that is the query) exists or not in the database. High CC value means that there exists the recurrence of the pattern in the database. Therefore, we could easily find the pattern. Table 3 indicates the maximum of cross-correlation between the query and reference windows.

Table 3: The maximum of cross-correlation between the query and reference windows.

Gap size	**Data set**							
	#1	#2	#3	#4	#5	#6	#7	#8
6%	0.88	0.92	0.58	0.78	0.99	1	0.91	1
7.50%	0.91	0.91	0.55	0.74	0.99	0.99	0.91	1
10%	0.94	0.87	0.5	0.67	0.98	0.99	0.91	1
12.50%	0.95	0.89	0.44	0.65	0.98	0.99	0.9	1
15%	0.95	0.85	0.4	0.65	0.98	0.99	0.9	1

#1-Airpassenger, #2-Beersales, #3-Google, #4-SP, #5-Co2 concentrations
#6-Mackey-Glass chaotic, #7-Phu Lien temperature, #8-water level

This result is fully interpreted: for 4 data sets including CO2 concentrations, Mackey-Glass chaotic series, Phu Lien temperature and water level, their cross-correlation

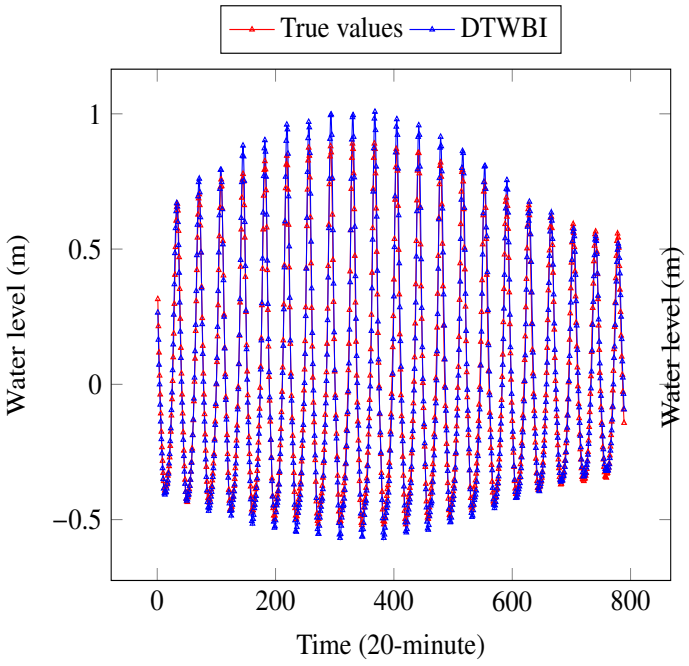

Figure 5: Visual comparison of imputed values of the proposed method with true values on water level series at position 23,282 with the gap size of 789.

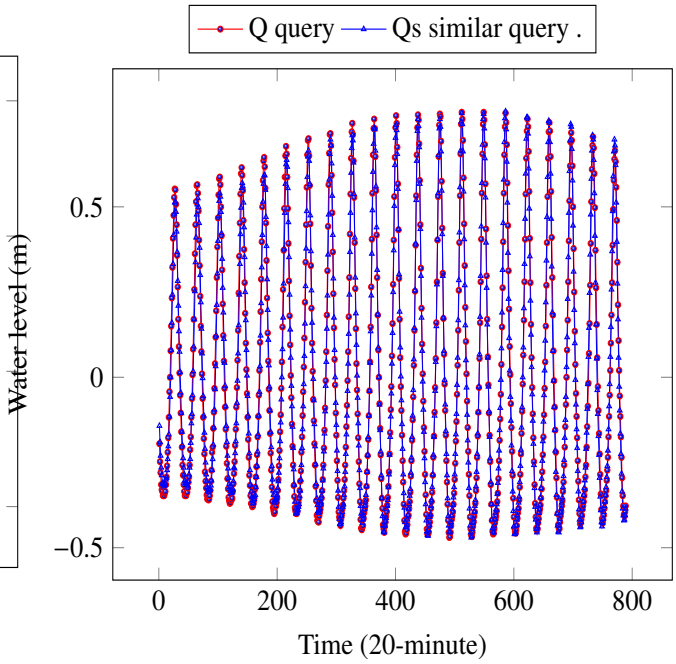

Figure 6: Visual comparison of the query with the similar window on water level series at position 23,282 with the gap size of 789.

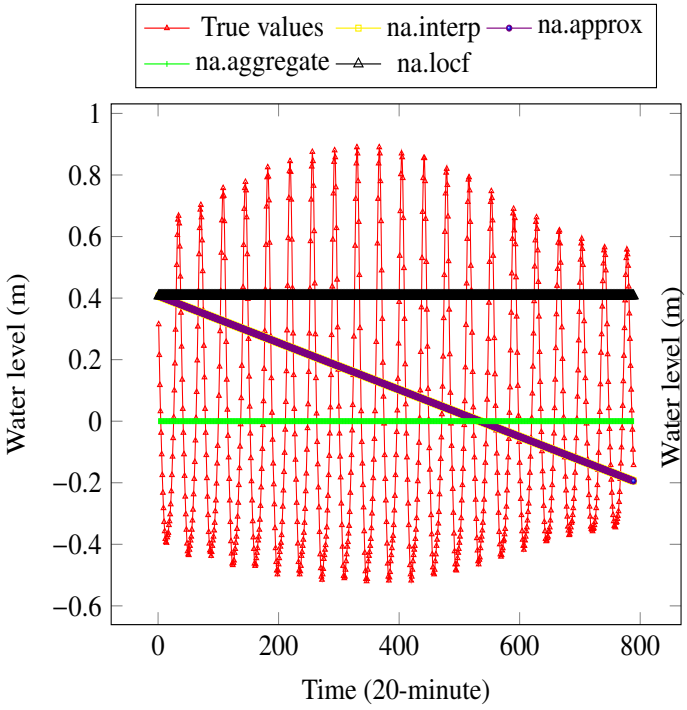

Figure 7: Visual comparison of imputed values of different methods with true values on water level series at position 23,282 with the gap size of 789.

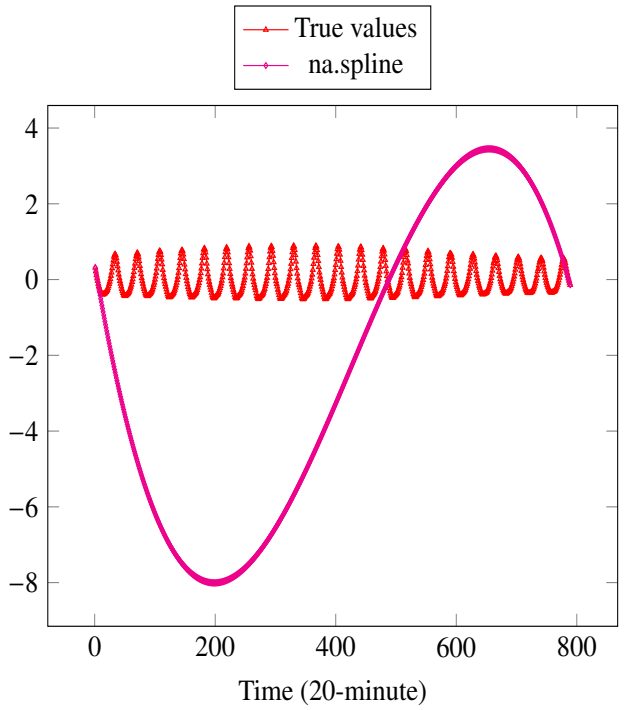

Figure 8: Visual comparison of imputed values of spline method with true values on water level series at position 23,282 with the gap size of 789.

between the query and reference windows are very high for each missing level (Table 3). This corresponds to the results in Table 2: the proposed method yields the highest similarity and the lowest NMAE, RMSE, FSD. It also means that the imputation values generated from DTWBI method are very close to the true ones. For Google (#3) and SP (#4) data sets, we see that CC are not high, that is why our approach does not well prove its ability. With Airpassenger data set (#1), when CC are greater than or equal to 0.94, the proposed method highlights better results than other methods. On Beersales data set (#2), in case of higher CC, DTWBI gives the best results in case of lower CC.

From these results, we can notice that the proposed method gives the best performance in case of high CC coefficient (> 0.9). Indeed, CC is an indicator that gives information about the pattern recurrence in the data. Based on this indicator, we can predict if one pattern may occur in the past or in the following data from the position we are considering. From the above analyses, we can see that our algorithm outperforms other imputation methods when data sets have high auto-correlation and cross-correlation, no trend, strong seasonality, and complex distribution, especially in case of large gap(s). High cross-correlation means that these data sets are recurrent, or in other words, these time series will repeat themselves over some periods. The drawback of this method is the computation time. The proposed algorithm may take a long time to find the imputation values when the size of the given data is large. The reason is the search for all possible sliding windows to find a reference window having the maximum similarity to the query.

6. Conclusion

In this paper, we have proposed a new imputation method for univariate time series data, namely DTWBI

method. This methodology has been tested using 8 data sets: Airpassenger, Beersales, Google, SP, Co2 concentrations, Mackey-Glass chaotic, Phu Lien temperature, and water level. The accuracy of imputation values by DTWBI is compared with 5 existing methods (na.interp, na.locf, na.approx, na.aggegate and na.spline) using 4 quantitative indicators (similarity, NMAE, RMSE and FSD). We also compare the visual performance of these methods. The experiments show that our approach gives better results than the other existing methods, and is the best robust method in case of time series having high cross-correlation and auto-correlation, large gap(s), complex distribution, and strong seasonality. However, the proposed framework is restricted to applications where the necessary assumption of recurring data in the time series is set up (high cross-correlation indicator), and it requires computation time for very large missing intervals. The present work will allow to extend the proposed approach to complete missing values of multivariate time series data in the future.

Acknowledgments

This work was kindly supported by the Ministry of Education and Training Vietnam International Education Development, the French government, the region Hauts-de-France in the framework of the project CPER 2014-2020 MARCO and the European Commission's H2020 program with the Joint European Research Infrastructure for Coastal Observations JERICO-Next.

References

Allison, P.D., 2001. Missing Data. volume 136 of *Quantitative Applications in the Social Sciences*. Sage Publication.

Bishop, C.M., 2006. Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Ceong, H.T., Kim, H.J., Park, J.S., 2012. Discovery of and recovery from failure in a costal marine usn service. Journal of Information and Communication Convergence Engineering 1.

Chiewchanwattana, S., Lursinsap, C., Henry Chu, C.H., 2007. Imputing incomplete time-series data based on varied-window similarity measure of data sequences. Pattern Recognition Letters 28, 1091–1103.

Crawford, S.L., Tennstedt, S.L., McKinlay, J.B., 1995. A comparison of anlaytic methods for non-random missingness of outcome data. Journal of Clinical Epidemiology 48, 209–219.

Deng, Y., Chang, C., Ido, M.S., Long, Q., 2016. Multiple Imputation for General Missing Data Patterns in the Presence of High-dimensional Data. Scientific Reports 6, 21689.

Gelman, A., Hill, J., Su, Y.S., Yajima, M., Pittau, M., Goodrich, B., Si, Y., Kropko, J., 2015. Mi: Missing Data Imputation and Model Checking.

Gómez-Carracedo, M., Andrade, J., López-Mahía, P., Muniategui, S., Prada, D., 2014. A practical comparison of single and multiple imputation methods to handle complex missing data in air quality datasets. Chemometrics and Intelligent Laboratory Systems 134, 23–33.

Hawthorne, G., Elliott, P., 2005. Imputing cross-sectional missing data: Comparison of common techniques. The Australian and New Zealand Journal of Psychiatry 39, 583–590.

Hyndman, R., Khandakar, Y., 2008. Automatic time series forecasting: the forecast package for r, used package in 2016. Journal of Statistical Software , 1–22URL: `http://www.jstatsoft.org/article/view/v027i03`.

Joseph, J.G., El-Mohandes, A.A.E., Kiely, M., El-Khorazaty, M.N., Gantz, M.G., Johnson, A.A., Katz, K.S., Blake, S.M., Rossi, M.W., Subramanian, S., 2009. Reducing Psychosocial and Behavioral Pregnancy Risk Factors: Results of a Randomized Clinical Trial Among High-Risk Pregnant African American Women. American Journal of Public Health 99, 1053–1061.

Junninen, H., Niska, H., Tuppurainen, K., Ruuskanen, J., Kolehmainen, M., 2004. Methods for imputation of

missing values in air quality data sets. Atmospheric Environment 38, 2895–2907.

Keogh, E.J., Pazzani, M.J., 2001. Derivative Dynamic Time Warping., in: Sdm, SIAM. pp. 5–7.

Lee, K.J., Carlin, J.B., 2010. Multiple Imputation for Missing Data: Fully Conditional Specification Versus Multivariate Normal Imputation. American Journal of Epidemiology 171, 624–632.

Lefebvre, A., 2015. MAREL Carnot data and metadata from Coriolis Data Centre. SEANOE. http://doi.org/10.17882/39754.

Liao, S.G., Lin, Y., Kang, D.D., Chandra, D., Bon, J., Kaminski, N., Sciurba, F.C., Tseng, G.C., 2014. Missing value imputation in high-dimensional phenomic data: Imputable or not, and how? BMC Bioinformatics 15, 346.

Little, R.J.A., Rubin, D.B., 2014. Statistical Analysis with Missing Data. John Wiley & Sons. Google-Books-ID: AyVeBAAAQBAJ.

Mackey, M.C., Glass, L., 1977. Oscillation and chaos in physiological control systems. Science (New York, N.Y.) 197, 287–289.

Moritz, S., Sardá, A., Bartz-Beielstein, T., Zaefferer, M., Stork, J., 2015. Comparison of different Methods for Univariate Time Series Imputation in R. arXiv preprint arXiv:1510.03924 .

Noor, N.M., Al Bakri Abdullah, M.M., Yahaya, A.S., Ramli, N.A., 2014. Comparison of Linear Interpolation Method and Mean Method to Replace the Missing Values in Environmental Data Set. Materials Science Forum 803, 278–281.

Phan, T.T.H., Caillault, E.P., Bigand, A., 2016. Comparative study on supervised learning methods for identifying phytoplankton species, in: 2016 IEEE Sixth International Conference on Communications and Electronics (ICCE), IEEE. pp. 283–288.

R Core Team, 2016. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing. Vienna, Austria. URL: `http://www.R-project.org/`.

Raghunathan, T.E., Lepkowski, J.M., Van Hoewyk, J., Solenberger, P., 2001. A multivariate technique for multiply imputing missing values using a sequence of regression models. Survey methodology 27, 85–96.

Raghunathan, T.E., Siscovick, D.S., 1996. A Multiple-Imputation Analysis of a Case-Control Study of the Risk of Primary Cardiac Arrest Among Pharmacologically Treated Hypertensives on JSTOR. Royal Statistical Society. Series C (Applied Statistics) 45, 335–352.

Rahman, S.A., Huang, Y., Claassen, J., Heintzman, N., Kleinberg, S., 2015. Combining Fourier and lagged k-nearest neighbor imputation for biomedical time series data. Journal of Biomedical Informatics 58, 198–207.

Rousseeuw, K., Caillault, E.P., Lefebvre, A., Hamad, D., 2013. Monitoring system of phytoplankton blooms by using unsupervised classifier and time modeling, in: 2013 IEEE International Geoscience and Remote Sensing Symposium-IGARSS, IEEE. pp. 3962–3965.

Royston, P., 2007. Multiple imputation of missing values: Further update of ice, with an emphasis on interval censoring. Stata Journal 7, 445–464.

Sakoe, H., Chiba, S., 1978. Dynamic Programming Algorithm Optimization for Spoken Word Recognition. IEEE transactions on acoustics, speech, and signal processing 16, 43–49.

Schafer, J., 1997. Analysis of Incomplete Multivariate Data. Chapman and Hall, London.

Shah, A.D., Bartlett, J.W., Carpenter, J., Nicholas, O., Hemingway, H., 2014. Comparison of random forest and parametric imputation models for imputing missing data using MICE: A CALIBER study. American Journal of Epidemiology 179, 764–774.

Spratt, M., Carpenter, J., Sterne, J.A.C., Carlin, J.B., Heron, J., Henderson, J., Tilling, K., 2010. Strategies for Multiple Imputation in Longitudinal Studies. American Journal of Epidemiology 172, 478–487.

Sterne, J.A.C., White, I.R., Carlin, J.B., Spratt, M., Royston, P., Kenward, M.G., Wood, A.M., Carpenter, J.R., 2009. Multiple imputation for missing data in epidemiological and clinical research: Potential and pitfalls. BMJ (Clinical research ed.) 338, b2393.

Stuart, E.A., Azur, M., Frangakis, C., Leaf, P., 2009. Multiple Imputation With Large Data Sets: A Case Study of the Children's Mental Health Initiative. American Journal of Epidemiology 169, 1133–1139.

Thoning, K.W., Tans, P.P., Komhyr, W.D., 1989. Atmospheric carbon dioxide at Mauna Loa Observatory. II - Analysis of the NOAA GMCC data, 1974-1985 94, 8549–8565.

Van Buuren, S., Boshuizen, H.C., Knook, D.L., others, 1999. Multiple imputation of missing blood pressure covariates in survival analysis. Statistics in medicine 18, 681–694.

Walter.O, Y., Kihoro, J.M., Athiany, K.H.O., W, K.H., 2013. Imputation of incomplete non- stationary seasonal time series data. Mathematical Theory and Modeling 3, 142–154.

Zeileis, A., Grothendieck, G., 2005. zoo: S3 infrastructure for regular and irregular time series, used package in 2016. URL: `https://www.jstatsoft.org/v014/i06`, doi:`10.18637/jss.v014.i06`.

Algorithm 1 DTWBI algorithm

Input: $x = \{x_1, x_2, \ldots, x_N\}$: incomplete time series
t: index of a gap (position of the first missing of the gap)
T: size of the gap
θ_cos: cosine threshold (≤ 1)
step_threshold: increment for finding a threshold
step_sim_win: increment for finding a similar window

Output: y - completed (imputed) time series

Step 1: Transform x to DDTW data $Dx = DDTW(x)$
Step 2: Construct a Q query - temporal window before the missing data $Q = Dx[t - T : t - 1]$
Step 3: Build a search database before the gap: $SDB = Dx[1 : t - 2T]$ and deleting all lines containing missing parameter $SDB = SDB \backslash \{dx_j, dx_j = NA\}$
Step 4: Find the threshold
$i \leftarrow 1$; $DTW\_costs \leftarrow NULL$
while $i <= length(SDB)$ **do**
    $k \leftarrow i + T - 1$
    Create a reference window: $R(i) = SDB[i : k]$
    Calculate global feature of Q and $R(i)$: gfQ, gfR
    Compute cosine coefficient: $cos = cosine(gfQ, gfR)$
    if $cos \geq \theta\_cos$ **then**
        Calculate DTW cost: $cost = DTW\_cost(Q, R(i))$
        Save the cost to DTW_costs
    end if
    $i \leftarrow i + step\_threshold$
end while
$threshold = min\{DTW\_costs\}$
Step 5: Find similar windows on the SDB
$i \leftarrow 1$; $Lop \leftarrow NULL$
while $i < length(SDB)$ **do**
    $k \leftarrow i + T - 1$
    Create a reference window: $R(i) = SDB[i : k]$
    Calculate global feature of Q and $R(i)$: gfQ, gfR
    Compute cosine coefficient: $cos = cosine(gfQ, gfR)$
    if $cos \geq \theta\_cos$ **then**
        Calculate DTW cost: $cost = DTW\_cost(Q, R(i))$
        if $cost < threshold$ **then**
            Save position of $R(i)$ to Lop
        end if
    end if
    $i \leftarrow i + step\_sim\_win$
end while
Step 6: Replace the missing values at the position t by vector after the Qs window having the minimum DTW cost in the Lop list.
return y - with imputed series