Supplementary material for the work titled "Similarity Measure Selection for Clustering Time Series Databases".

Usue Mori, Alexander Mendiburu, Jose A. Lozano.
Intelligent Systems Group (ISG),
Department of Computer Science and Artificial Intelligence,
University of the Basque Country UPV/EHU.
Address: Manuel de Lardizabal 1, 20018 Donostia-San Sebastian, Spain.

Abstract

In this document the generation process of the synthetic databases used in the work titled "Similarity Measure Selection for Clustering Time Series Databases" is explained in detail. By using this information and the code attached, the readers can easily generate the databases used in this study. Furthermore, the explanations and code enable the generation of new databases with different characteristics that can be used in other research projects.

1 Overall synthetic database generation process

In this first section, the overall process for the generation of the synthetic databases used in the work titled "Similarity Measure Selection for Clustering Time Series Databases" is introduced. It must be noted that these databases are designed to be used for time series clustering tasks. As such, they contain several underlying groups or clusters, which are intentionally and artificially generated. This fact will be one of the main points of the generation process and will be explained in the following paragraphs.

Database type	**Noise level**	**Outliers level**	**Shift level**	**Warp level**
Armas ($k^* = 8$) Synthetic control ($k^* = 6$)	None=0 Low={1,2} Med={3,4} High={5,6}	None=0 Low={1,2} Med={3,4} High={5,6}	None=0 Low={1,...,5} Med={6,...,12} High={13,...,30}	None=0
Sines ($k^* = 5$) Rational ($k^* = 4$) Seasonal ($k^* = 4$)	None=0 Low={1,2} Med={3,4} High={5,6}	None=0 Low={1,2} Med={3,4} High={5,6}	None=0 Low={1,...,5} Med={6,...,12} High={13,...,18}	None=0
CBF ($k^* = 3$)	None=0 Low={1,2,3} High={3,4,5}	None=0 Low={1,2,3} High={3,4,5}	None=0 Low={1,...,5} High={6,...12}	None=0 Low={1,...,5} High={6,...,10}
Two Patterns ($k^* = 4$) Kohler & Lorenz ($k^* = 5$)	None=0 Low={1,2,3} High={3,4,5}	None=0 Low={1,2,3} High={3,4,5}	None=0 Low={1,...,5} High={6,...,10}	None=0 Low={1,...,5} High={6,...,8}

Table 1: Characteristic options for the generation of the synthetic databases.

The synthetic databases generated for this work can be divided into 8 distinct groups (see second column, **'Database type'** in Table 1). For each type of database various different examples have been generated by modifying the level of noise, the level of outliers, the mean level of shift and the mean level of local warp within the ranges shown in Table 1. In order to avoid homogeneity, only one example database for each combination of categories has been generated for each database type. Furthermore, the databases with 0 values for all options have been discarded because they are too simple and unrealistic. An example of this generation process for the databases of type CBF is shown in Algorithm 1.

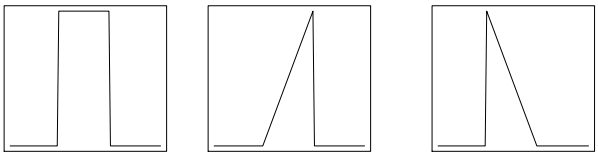
Algorithm 1 Generation of the synthetic databases of type CBF.

1: **for** $noise$ in $\{none, low, high\}$ **do**
2: **for** $outliers$ in $\{none, low, high\}$ **do**
3: **for** $shift$ in $\{none, low, high\}$ **do**
4: **for** $warp$ in $\{none, low, high\}$ **do**
5: **if** $noise \neq none$ **or** $outlier \neq none$ **or** $shift \neq none$ **or** $warp \neq none$ **then**
6: $noise\_level$ $\leftarrow$ value randomly chosen inside $noise$ category.
7: $outlier\_level$ $\leftarrow$ value ranmdoly chosen inside $outlier$ category.
8: $shift\_level$ $\leftarrow$ value randomly chosen inside $shift$ category.
9: $warp\_level$ $\leftarrow$ value randomly chosen inside $warp$ category.
10: $dimension$ $\leftarrow$ value randomly chosen from the second column of Table 2.
11: $proportions$ $\leftarrow$ value randomly chosen from the third column of Table 2.
12: Generate and save database of type CBF with selected options.
13: **return** 80 databases of type CBF

If this same process is repeated for all the database types, it results in a total of 555 synthetic databases ($2 \times 4 \times 4 \times 4 \times 1 + 3 \times 4 \times 4 \times 4 \times 1 + 1 \times 3 \times 3 \times 3 \times 3 + 2 \times 3 \times 3 \times 3 \times 3 - 8$). Nevertheless, in order to be able to apply Algorithm 1, some additional points must be specified.

First, the meaning of the variables $dimension$ and $proportions$ must be explained (see lines 10 and 11 of Algorithm 1). The first one represents the dimension of the database and is specified by the number of series that it contains and their respective length ($N \times L$). To define the second variable we must recall that we are working in a clustering environment. Because of this, each synthetic dataset will be generated is such a way that it will contain a predefined and previously known set of clusters. The series belonging to the same cluster will have similar shapes. In this context, the $proportion$ variable represents the proportion of series belonging to each cluster.

Once this is explained, a method to generate a database of a given type and with some specific characteristics must be defined (see line 12 of Algorithm 1). Each type of synthetic database included in this study is defined by k^* predefined shapes, each shape representing one cluster. For example, in the classical Cylinder-Bell-Funnel (CBF) database [Keogh et al., 2011], the three shapes that can be seen in Figure 1 are used to represent three different clusters. For the details about the basic shapes of the other types of synthetic database see Section 2.



(a) Cylinder. (b) Bell. (c) Funnel.

Figure 1: Shapes that define the CBF synthetic database.

To create a synthetic database of a certain type (e.g., CBF) and with some specific characteristics, the dimension and the proportion of series in each cluster are chosen randomly from the set of predefined options (see Table 2). Based on this, the k^* shapes that define that type of synthetic database (the shapes in Figure 1 in the case of CBF) are replicated accordingly. Next, the noise, outliers, shift and warp will be added to the database in the following manner:

- **Noise:** The noise is introduced separately in each series of the database by adding random values issued from a normal distribution of mean 0. The standard deviation (σ) of this distribution is defined by the level of noise normalized by the maximum (max) and minimum (min) values of the series:

$$\sigma = \frac{\text{noise level}}{max - min} \tag{1}$$

- **Outliers:** The introduction of outliers will be carried out independently for each time series in the database. The selected outlier level will represent the proportion of points in the series that will become outliers. Given a specific outlier level, the corresponding number of points are selected randomly from the series and interchanged with points randomly chosen from other series in order to convert them into outliers.

- **Shift:** Contrary to previous features, the introduction of shift in a time series database can not be carried out separately for each time series because it is a global variable. Each series must be shifted in a specific manner so that, overall, a pre-specified mean level of shift is obtained. The proposed solution is to shift each time series X_i following a random variable P_i that takes random integer values in the interval $[-m, m]$, m being a positive integer. If this procedure is followed, the mean level of shift (sh) introduced in the database can be quantified by calculating the expectation of the mean phase difference between all pairs of series in the database:

$$sh = \frac{2}{N(N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \mathbb{E}(|P_i - P_j|) = \frac{4(m+1)m}{3(2m+1)} \tag{2}$$

  where N is the total number of series in the database. Now, we are interested in the inverse problem: given a level of mean shift (sh), we want to find the maximum value m to which we can shift the series. This can be obtained by simply calculating the inverse of Equation 2:

$$m = \frac{3 \cdot sh - 2 + \sqrt{9 \cdot sh^2 + 4}}{4} \tag{3}$$

  With this equation and the procedure explained above, the shift can be introduced in a controlled manner in synthetic time series databases. The only inconvenience of Equation 3 is that, for most values of sh, this inverse function does not return integer values. This is solved by simply truncating the result.

- **Warp or local scaling**: Given the difficulty of introducing warp in general time series databases, it has only been considered for the CBF [Keogh et al., 2011], Two Patterns [Geurts, 2002] and Kohler & Lorenz [Köhler and Lorenz, 2005] database types, where it can be done in a simple way. At least one of the shapes in these types of synthetic databases is a piecewise function and, in this case, the local scaling is introduced by modifying the limits of the pieces in both directions in a random way and inside a given interval $[-m, m]$, where m is chosen from the options shown in the last column of Table 1.

The code used to generate the databases and the databases themselves are available together with this document in the Downloads section of our website [1].

[1] http://www.sc.ehu.es/isg

2 Additional specifications of the synthetic database generation

In this section, some additional specifications needed to completely determine the generation process of the synthetic databases are given. To begin with, we recall that in the generation of the synthetic databases the dimension of the database and the proportion of series in each cluster were chosen randomly from a set of options. In Table 2 we summarize these options.

Database type	**Dimension**	**Proportions**
Armas (K=8)	70x70, 100x70, 150x70, 200x70, 300x70, 500x70, 70x100, 100x100, 150x100, 200x100, 300x100, 500x100, 70x150, 100x150, 150x150, 200x150, 300x150, 70x200, 100x200, 150x200, 200x200, 300x200, 70x300, 100x300, 150x300, 200x300, 70x500, 100x500	(1/8,1/8,1/8,1/8,1/8,1/8,1/8,1/8) (1/4,1/8,1/8,1/16,3/16,1/8,1/8,0) (1/5,1/5,0,0,1/5,0,1/5,1/5) (1/4,0,1/3,1/8,1/8,0,1/6,0)
Synthetic control (K=6)		(1/6,1/6,1/6,1/6,1/6,1/6) (1/6,1/12,1/12,1/6,1/4,1/4) (1/3,1/3,1/12,1/12,1/12,1/12)
Sines (K=5) Kohler & Lorenz (K=5)		(1/5,1/5,1/5,1/5,1/5) (1/10,1/5,1/10,2/5,1/5) (1/15,2/5,1/15,2/5,1/15) (2/5,1/10,1/5,1/10,1/5)
CBF (K=3)		(1/3,1/3,1/3) (1/6,1/3,1/2) (2/3,1/6,1/6)
Two Patterns (K=4) Rational (K=4) Seasonal (K=4)		(1/4,1/4,1/4,1/4) (1/3,1/3,1/6,1/6) (1/8,1/4,1/2,1/8)

Table 2: Possible values for the $dimension$ and $proportion$ variables.

Furthermore, in order to fully describe the different synthetic database types (see 'Database Type' in Table 1) the shapes that define each of them are specified:

- **ARMAs:** 8 series obtained from different initializations of an ARMA(3,2) process with coefficients AR=(1,-0.24,0.1) and MA=(1,1.2) conform this synthetic database.
- **Synthetic control:** This time series database is an example of a synthetic database that was first introduced by Pham and Chan [1998]. It is based on 6 basic shapes:

$$f_1(t) = 80 + r(t)$$
$$f_2(t) = 80 + 15\sin(\frac{2\pi t}{T})$$
$$f_3(t) = 80 + 0.4t$$
$$f_4(t) = 80 - 0.4t$$
$$f_5(t) = \begin{cases} 80 & t \leq \lfloor \frac{L}{2} \rfloor \\ 90 & \lfloor \frac{L}{2} \rfloor < t \end{cases}$$
$$f_6(t) = \begin{cases} 90 & t \leq \lfloor \frac{L}{2} \rfloor \\ 80 & \lfloor \frac{L}{2} \rfloor < t \end{cases}$$

  where r(t) is a random value issued from a $\mathcal{N}(0,3)$ distribution, L is the length of the series and T is the period and is defined as a third of the length of the series.

- **Sines:** This type of synthetic database is defined by 4 different shapes that are based on the function $f(t) = sin(\frac{2\pi t}{T}) \cdot r$, where r is a random number between 1 and 1.1 and $t = 1, ..., L$. The first shape is

obtained by generating instances of f itself. The second shape is obtained by fitting Local Polynomial Regressions (LOESS) to instances of f using polynomials of degree 1 and a smoothing parameter of 0.25. To build the third shape, a sample of the f function is taken, but the values higher than τ are modified into a constant value of τ, τ being a random number between 0.9 and 0.99. The fourth shape is the negative counterpart of the previous case. Finally, the fifth shape takes an instance of f and adds a random number between 0.2 and 0.3 to a part of the series. The objective of introducing this synthetic database type is to create databases with very similar clusters.

- **Rational:** This database type is generated following the same idea as in the previous case and consists of 4 shapes. The first shape is defined by the values of $f(x) = \frac{x}{x^2+1} \cdot r$ in L equidistant points in the interval $[-20, 20]$, r being a random number between 1 and 1.1 and L the length of the series. The second shape is defined by the function $g(x) = \frac{x}{x^2+2} \cdot r$, also in the interval $[-20, 20]$. The third is a smooth LOESS approximation of f with polynomials of degree 1 and a smoothing parameter of 0.4, and the fourth is a cubic spline interpolation.

- **Seasonal:** This type of database consists of 4 shapes based on adding sinusoidal functions:

$$f_1(t) = \sin(\frac{2\pi t}{T}) + \frac{1}{2}\sin(\frac{4\pi t}{T} - 2) + \frac{1}{3}\sin(\frac{4\pi t}{T} - 2) + \frac{1}{3}\cos(\frac{8\pi t}{T} - 2) + \sin(\frac{10\pi t}{T} - 6)$$
$$f_2(t) = \frac{3}{2}\sin(\frac{2\pi t}{T}) + \frac{1}{2}\cos(\frac{12\pi t}{T} - 1) + \frac{1}{6}\cos(\frac{4\pi t}{T} - 7) + \sin(\frac{4\pi t}{T} - 9)$$
$$f_3(t) = 1.2\cos(\frac{6\pi t}{T} + 7) + 0.9\sin(\frac{2\pi t}{T} - 2) + \frac{1}{3}\cos(\frac{2\pi t}{7T})$$
$$f_4(t) = 0.24\sin(\frac{4.4\pi t}{T} + 7) + \frac{1}{2}\cos(\frac{2\pi t}{T} - 5) + \frac{1}{3}\cos(\frac{4\pi t}{7T})$$

  where T is a half of the series length and $t = 1, ..., L$. The objective of mixing many sinusoidal forms is to obtain time series with many peaks.

- **CBF:** This type of synthetic database is also available in the UCR repository. Three basic shapes are defined as follows:

$$f_1(t) = \begin{cases} 0 & t \leq a \\ 6 + r(t) & a < t \leq b \\ 0 & t \geq b \end{cases} \qquad f_2(t) = \begin{cases} 0 & t \leq a \\ (6 + r(t)) \cdot \frac{t-a}{b-a} & a < t \leq b \\ 0 & t \geq b \end{cases}$$

$$f_3(t) = \begin{cases} 0 & t \leq a \\ (6 + r(t)) \cdot \frac{b-t}{b-a} & a < t \leq b \\ 0 & t \geq b \end{cases}$$

  where a is initially defined as $\lfloor \frac{L}{3} \rfloor$ and b as $\lfloor \frac{2L}{3} \rfloor$, being L the length of the series.

- **Two Patterns:** This type of synthetic database was introduced by Geurts [2002] and an example is included in the UCR repository. The idea is two combine two patterns us and ds as follows:

$$f_1(t) = \begin{cases} r(t) & 0 \leq t \leq t_1 \\ us(t-t_1, l_1) & t_1 < t \leq t_1 + l_1 \\ r(t) & t_1 + l_1 < t \leq t_2 \\ us(t-t_2, l_2) & t_2 < t \leq t_2 + l_2 \\ r(t) & t \geq t_2 + l_2 \end{cases} \qquad f_2(t) = \begin{cases} r(t) & 0 \leq t \leq t_1 \\ us(t-t_1, l_1) & t_1 < t \leq t_1 + l_1 \\ r(t) & t_1 + l_1 < t \leq t_2 \\ ds(t-t_2, l_2) & t_2 < t \leq t_2 + l_2 \\ r(t) & t \geq t_2 + l_2 \end{cases}$$

$$f_3(t) = \begin{cases} r(t) & 0 \leq t \leq t_1 \\ ds(t-t_1, l_1) & t_1 < t \leq t_1 + l_1 \\ r(t) & t_1 + l_1 < t \leq t_2 \\ us(t-t_2, l_2) & t_2 < t \leq t_2 + l_2 \\ r(t) & t \geq t_2 + l_2 \end{cases} \qquad f_4(t) = \begin{cases} r(t) & 0 \leq t \leq t_1 \\ ds(t-t_1, l_1) & t_1 < t \leq t_1 + l_1 \\ r(t) & t_1 + l_1 < t \leq t_2 \\ ds(t-t_2, l_2) & t_2 < t \leq t_2 + l_2 \\ r(t) & t \geq t_2 + l_2 \end{cases}$$

where,

$$us(t,l) = \begin{cases} -5 & 0 \leq t \leq \frac{l}{2} \\ 5 & t_1 < t \leq t_1 + l_1 \end{cases} \qquad ds(t,l) = \begin{cases} 5 & 0 \leq t \leq \frac{l}{2} \\ -5 & t_1 < t \leq t_1 + l_1 \end{cases}$$

where, t_1 is initially set in $\lfloor \frac{L}{3} \rfloor$, t_2 in $\lfloor \frac{2L}{3} \rfloor$, l_1 and l_2 both take a value of $0.1 \cdot L$ and $r(t)$ is a random value extracted from a $\mathcal{N}(0,1)$ distribution.

- **Kohler & Lorenz:** This last synthetic database was created by making some modifications on the synthetic series presented in Köhler and Lorenz [2005]. The following 5 shapes are defined:

$$f_1(t) = 13 \cdot r(t) \sin(5 \cdot r(t) \frac{t}{T}) \qquad f_2(t) = \begin{cases} \frac{-t^2}{100} & t \leq \lfloor \frac{L}{2} \rfloor \\ \frac{t^2}{530} & \lfloor \frac{L}{2} \rfloor > t \end{cases}$$

$$f_3(t) = \begin{cases} 5 & t \leq \lfloor \frac{L}{3} \rfloor \\ -12 & \lfloor \frac{L}{3} \rfloor < t \leq \lfloor \frac{2L}{3} \rfloor \\ 8 & t \geq \lfloor \frac{2L}{3} \rfloor \end{cases} \qquad f_4(t) = \begin{cases} 15 \cdot r(t) \sin(15 \cdot r(t) \frac{t}{T}) & t \leq \lfloor \frac{L}{8} \rfloor \\ r(t) \sin(7 \cdot r(t) \frac{t}{T}) & \lfloor \frac{L}{8} \rfloor < t \leq \lfloor \frac{2L}{8} \rfloor \\ 10 & \lfloor \frac{2L}{8} \rfloor < t \leq \lfloor \frac{3L}{8} \rfloor \\ -2 & \lfloor \frac{3L}{8} \rfloor < t \leq \lfloor \frac{4L}{8} \rfloor \\ \frac{(10t/L)^2}{2} & \lfloor \frac{4L}{8} \rfloor < t \leq \lfloor \frac{5L}{8} \rfloor \\ \frac{-(2t/L)^2}{2} & t \geq \lfloor \frac{5L}{8} \rfloor \end{cases}$$

$f_5(t) =$ series generated from an ARMA(3,2) process with parameters AR=(1,-0.24,0.1) and MA=(1,1.2)

where L is the length of the series, T is 10% of the length of the series and $r(t)$ is a random number obtained from a $\mathcal{N}(0, 0.1)$ distribution.

References

P. Geurts. *Contributions to decision tree induction: bias/variance tradeoff and time series classification.* PhD thesis, University of Liege, Belgium., 2002.

E. Keogh, Q. Zhu, B. Hu, Hao Y., X. Xi, and C. A. Wei, L. Ratanamahatana. The UCR Time Series Classification/Clustering Homepage, 2011. URL `www.cs.ucr.edu/~eamonn/time\_series\_data/`.

T. Köhler and Dirk Lorenz. A comparison of denoising methods for one dimensional time series. Technical report, 2005. URL `http://www.math.uni-bremen.de/zetem/DFG-Schwerpunkt/preprints/orig/lorenz20051dreport.pdf`.

D. T. Pham and A. B. Chan. Control chart pattern recognition using a new type of self-organizing neural network. *Proceedings of The Institution of Mechanical Engineers Part I-journal of Systems and Control Engineering*, 212(2):115–127, 1998.