Chaos, Solitons and Fractals 142 (2021) 110326

Contents lists available at ScienceDirect

Chaos, Solitons and Fractals

Nonlinear Science, and Nonequilibrium and Complex Phenomena

journal homepage: www.elsevier.com/locate/chaos

Temporal gap statistic: A new internal index to validate time series clustering

Rosana Guimarães Ribeiro*, Ricardo Rios

Department of Computer Science, Federal University of Bahia, Brazil

ARTICLE INFO

Article history:
Received 5 June 2020
Revised 4 September 2020
Accepted 28 September 2020
Available online 24 October 2020

Keywords:
Time series
Internal index
Clustering validity

ABSTRACT

Unsupervised Machine Learning techniques have been developed to find out structures in datasets without considering any prior information. In such a context, the main challenge is to confirm whether the obtained structure indeed contains relevant data patterns. Aiming at solving this issue, there are several validation indexes proposed under different categories (e.g. internal, external, and relative) that allow to, for example, compare clustering algorithms or define the best parameter configurations. However, most of those indices are applied to data characterized for being collected in an independent and identically distributed manner. Thus, after performing a Systematic Literature Review, we noticed there are few researches investigating validation indexes specifically designed to deal with time-dependent data. The absence of researches for such context has motivated this work that was devoted to developing a new internal index based on Gap Statistic. Our index supports the estimation of the optimal number of clusters in a dataset only composed of time series. To reach this goal, we performed three important modifications in Gap Statistic: i) the use of a measure to calculate the distance between time series; ii) the adoption of a clustering method based on medoid; and iii) the modeling of time series in phase space using Dynamical System tools. Our results emphasize the importance of the proposed index, by accurately clustering sets of chaotic time series.

1. Introduction

The significant amount of data currently produced by companies and users in general has motivated the development of Machine Learning (ML) techniques, which aim at inducing hypothesis to learn new information from historical experience [1–3]. Although there are several paradigms to support such hypothesis induction, in this work, we are focused on unsupervised learning which provides methods to extract patterns from data without taking into account any prior information provided by, for example, domain specialists. Once no information is known in advance, but the data features, the main challenge is to confirm whether the obtained structure indeed contains relevant patterns [3,4].

This challenge is tackled by using validity criteria [5], which implement indices to test and evaluate the quality of the obtained structures. As discussed in [3–6], such criteria are organized into three categories: external, relative, and internal. External criteria analyze structures produced by ML algorithms to confirm any previous hypothesis about the data. Relative criteria are widely used to compare different algorithms or parameters. Finally, internal criteria are applied to identify the best number of groups present in structures produced by ML algorithms. After analyzing researches published in the literature, we noticed most of clustering and validity algorithms were developed assuming that the data gathering information happens in an independent and identically distributed (iid) fashion. However, when data is characterized by temporal dependencies, i.e., the value of a current observation is related to one or more past values, then traditional algorithms may not be useful. The development and adaptation of algorithms, specifically designed to deal with time-dependent data, such as time series, have been proposed by several researchers. For example, Berndt and Clifford [7] have presented the Dynamic Time Warping (DTW) method to calculate the distance between temporal data, thus making it possible to replace Minkowski-based distance usually adopted by clustering algorithms and allowing their usage with temporal data.

However, we realized that the same effort has not been dedicated to evaluating the quality of clusters obtained on temporal data. This limitation has motivated our work, which aims at designing a new validity approach. Our research was built on top of the Gap Statistic approach, which is commonly used to cluster iid data. Briefly, such an approach compares an appropriate null

* Corresponding author.
E-mail addresses: rosana.guimaraes08@gmail.com (R.G. Ribeiro), ricardoar@ufba.br (R. Rios).

https://doi.org/10.1016/j.chaos.2020.110326
0960-0779/© 2020 Elsevier Ltd. All rights reserved.

distribution as reference with intra-cluster dispersion calculated on partitions produced by clustering algorithms. In that comparison, the main step is the creation of synthetic datasets produced by using, for example, a uniform distribution in a Monte Carlo method.

Based on that strategy, we have created the Temporal Gap Statistic approach by performing three modifications on the original Gap Statistic. The first one is related to the distance used to compare pairs of data instances. As a natural choice, we have replaced the euclidean distance by DTW. We use this distance not only in the clustering algorithm but also in the dispersion between partitions and the null reference. Secondly, we have used the K-medoid algorithm in place of K-means. Although both algorithms have similar behavior, the first one is more recommended when the distance measure is not capable of assuring the triangular inequality property. As discussed in this manuscript, this situation can lead to empty clusters, which is not conceptually accepted in unsupervised learning. Finally, our main contribution is to transform the data from the time domain to the phase space using Dynamical System tools. This transformation allows to better model nonlinear relationships among data observations, once attractors are better understood when the data is unfolded into a high-dimensional space.

The remaining of this manuscript is organized as follows: Section 2 introduces a theoretical foundation on Unsupervised Machine Learning; In Section 3, we present a review on related work published in the literature; Section 4 details our approach; the experimental setup and our results are presented in Section 5 and 6, respectively; and, finally, we have drawn our conclusion and presented our final remarks in Section 7.

2. Unsupervised machine learning methods

Unsupervised Machine Learning methods, or simply Clustering methods, were designed to perform an exploratory data analysis when no information, except the feature space, is provided by specialists [4]. As stated in [5], there is no single and precise definition on clustering. In general, algorithms are implemented aiming at organizing data in partitions (or structures) by minimizing distances among objects from the same group (intra-cluster) and by maximizing distances among different groups (inter-cluster).

The most well-known clustering methods are based on partitional and hierarchical approaches, whose mathematical definition starts stating a set of input patterns $X = \{x_1, \dots, x_j, \dots, x_N\}$ with $x_j = (x_{j1}, x_{j2}, \dots, x_{jd}) \in \mathbb{R}_d$, in which every measure x_{ji} is called feature (attribute, dimension, or variable) [4,5]. Thus, such approaches are defined as:

1. Hard partitional clustering attempts to seek a K-partition of X, $C = \{C_1, \dots, C_K\} (K \leq N)$, such that
    - $C_i \neq \emptyset, \forall i = \{1, \dots K\}$
    - $\bigcup_{i=1}^{k} C_i = X$
    - $C_i \cap C_j = \emptyset$, $\forall i, j = \{1, \dots, K\}, i \neq j$
2. Hierarchical clustering attempts to construct a tree-like, nested structure partition of X,$H = \{H_1, \dots, H_Q\} (Q \leq N)$, such that $C_i \in H_m$,$C_j \in H_l$, and $m > l$, imply $C_i \subset C_j$ or $C_i \cap C_j = \emptyset, \forall i, j \neq i, m, l = 1, \dots, Q$.

According to the literature, most of clustering methods assumes the input patterns X are characterized for being produced/collected in an independent and identically distributed (iid) manner, i.e., there is no temporal dependency among their observations (x_j). It is worth emphasizing this is a normal assumption in ML, specially in supervised methods, which is used to provide learning guarantees [8]. However, when the data is characterized by temporal dependencies, such as time series, the computation of intra- and inter-cluster distances must be accordingly adapted as discussed in the next section.

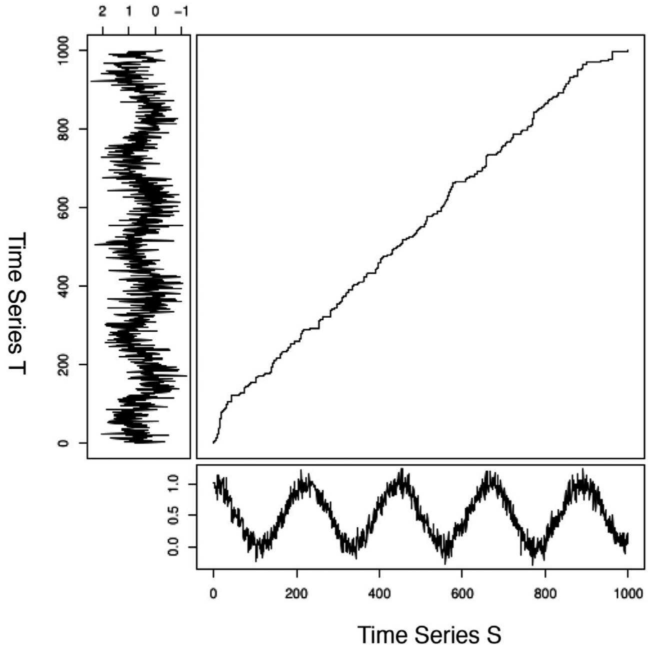


Fig. 1. An illustrative example of a warping path calculated between a sinusoidal time series with low-noise influences (X-axis) and another sinusoidal time series with high-noise influences (Y-axis). A perfect match between two time series produces a warping path with a straight diagonal line.

2.1. Distance measures

The most important step during the clustering process is the application of distance measures. Such measures are responsible for determining whether two objects, by comparing their features, might be placed into the same group. A distance widely used by clustering algorithms is Minkowski, which is a metric characterized by low computational complexities ($O(n)$). In the context of data with temporal dependencies, the main inconveniences of this metric are the requirement of the same number of features among objects and the need for perfect alignment between the analyzed time series. In summary, series must present the same length and a small displacement between them can completely affect results.

Aiming at overcoming these situations, researchers proposed the Dynamic Time Warping (DTW) method [9], which has been successfully applied to compare time series, specially for not requiring any parameter configuration. Towards a better understanding of this method, let $x_j = (x_{j1}, x_{j2}, \dots, x_{jn}) \in \mathbb{R}_n$ and $x_k = (x_{k1}, x_{k2}, \dots, x_{km}) \in \mathbb{R}_m$ be two time series with different sizes n and m, respectively. DTW algorithm starts creating a matrix $n \times m$, where each cell corresponds to an alignment between observations of x_j and x_k. Warping Path (W) maps, or aligns, elements of x_j and x_k, so that the distance between their observations is minimized [9,10] as shown Eq. (1), in which dist is a distance function between two observations. Fig. 1 exemplifies the warping path between two noisy time series created after adding uniformly distributed noises, with different values of standard deviation, to observations produced from a sine function. In terms of computational complexity, the DTW algorithm based on dynamic programming is $O(nm)$ [9,11,12].

$$DTW(x_j, x_k) = \sqrt{\text{dist}(x_{jn}, x_{km})} \tag{1}$$

$$\begin{aligned} &\text{dist}(x_{ji}, x_{kl}) \\ &\quad = (x_{ji}, x_{kl})^2 + \min \begin{cases} \text{dist}(x_{j(i-1)}, x_{kl}) \\ \text{dist}(x_{ji}, x_{k(l-1)}) \\ \text{dist}(x_{j(i-1)}, x_{k(l-1)}) \end{cases}, \quad i \in \{1, \dots, n\}, \quad l \in \{1, \dots, m\} \end{aligned} \tag{2}$$

As the reader may notice, our goal in this section is not to present an extensive review of distance methods. In turn, we focused our attention on DTW, which was considered in our proposed approach. For more information on this topic, we recommend reading the following manuscripts. [11,12].

2.2. Validity indexes

After selecting a distance method and a clustering algorithm, which can be based on partitional or hierarchical approach, the produced partition must be assessed to check whether a valid structure was retrieved. In this sense, cluster validation provides quantitative and objective methods to perform this task. As discussed in [4,5], such methods can be organized in three categories of criteria: external, relative, and internal. Some authors consider relative criteria as part of the internal ones. In this work, we use the taxonomy presented by [5] that is widely adopted in the literature.

2.2.1. External criteria

External criteria measure the clustering performance by using a pre-defined structure. In summary, it compares the degree of correspondence between the estimated clusters and category labels assigned in advance [4]. Aiming at better understanding these criteria, let P be a known (expected) partition of a dataset X, composed of N observations, and C be a clustering structure provided from a clustering algorithm. By taking into account pairs of inputs x_i and x_j of X, there are four different cases on how they are placed in C and P[5]:

- Case 1: x_i and x_j belong to the same clusters of C and the same category of P.
- Case 2: x_i and x_j belong to the same clusters of C but different categories of P.
- Case 3: x_i and x_j belong to different clusters of C but the same category of P.
- Case 4: x_i and x_j belong to different clusters of C and different category of P.

Correspondingly, the cases 1, 2, 3 and 4 are denominated as a,b,c and d, respectively, being $M = a + b + c + d$. Thereby, through the relationship between the different cases, it is possible to determine some external indices commonly used to measure the correspondence between C and P[5]:

- *Rand Index*

$$R = \frac{(a+d)}{M} \tag{3}$$

- *Jaccard coefficient*

$$J = \frac{a}{(a+b+c)} \tag{4}$$

- *Fowlkes and Mallows Index*

$$FM = \sqrt{\frac{a}{a+b}\frac{a}{a+c}} \tag{5}$$

- Γ *statistics*

$$\Gamma = \frac{Ma - m_1 m_2}{\sqrt{m_1 m_2 (M - m_1)(M - m_2)}} \tag{6}$$

where $m_1 = a + b$ and $m_2 = a + c$.

2.2.2. Relative criteria

Relative criteria perform cluster analyses when no information is previously known. They are usually adopted to compare clustering results generated by either different algorithms or the same algorithm with different parametrization [5]. For example, if one needs to find out the best number of groups, he/she runs multiple execution of an algorithm varying from 2 to k and checks which one provides the best index. In this sense, there exists several relative indexes in literature, such as, Calinski-Harabasz Index (VRC), Dunn's Index, and Davies-Bouldin Index. However, the most adopted index is Silhouette [13].

This index is based on geometric measures of compaction and separation groups (Eq. (7)). To understand this index, let x_j be an input/object that was placed into a given group $p \in \{1, \ldots, k\}$. Then, the mean distance among this object and all other objects from p is defined as $a(j, p)$. Similarly, the mean distance among this object and all objects placed into another group q,$q \neq p$, is represented by $d_{j,q}$. Finally, let $b(j)$ be the minimum distance between x_j and all groups, without considering p which is the cluster where it belongs, i.e., $\min\{d_{j,q}, \forall q \in \{1, \ldots, k\},\ q \neq p\}$. In summary, b_j represents the mean distance between x_j and its closest cluster [13].

$$s(x_j) = \frac{b(j) - a(j,p)}{\max\{a(j,p), b(j)\}} \tag{7}$$

In this case, the higher the value of $s(x_j)$ is, the better its allocation is. To analyze all inputs, it is commonly considered the mean Silhouette which is defined as Eq. (8). In this case, the best partition is achieved when S_μ is maximized, i.e., by minimizing intra-cluster distance $a(j, p)$ and maximizing inter-cluster distance $b(j)$.

$$S_\mu = \frac{1}{N}\sum_{j=1}^{N} s(x_j) \tag{8}$$

2.2.3. Internal criteria

Internal criteria also evaluate clustering structure without any external information about the analyzed data [5]. Moreover, they do not require several test with different algorithms and/or parametrization until finding the best clustering structure.

The most well-known approaches are Cophenetic Correlation Coefficient, used to validate hierarchical clustering structures [5], and Gap Statistic [14], which is used to validate partitional and hierarchical clustering structures.

Gap Statistic compares the within-cluster dispersion, provided by any clustering algorithm, and the expected dispersion under an appropriate null distribution as reference [14].

In order to better understand this approach, we reproduce in this section original work published in [14]. In this sense, consider Fig. 2(a) that illustrates two cloud of points in a 2-dimensional space. Gap Statistic starts running a clustering algorithm, such as Partitional or Hierarchical, by varying the total number of clusters in the range $k = \{1, 2, 3, \ldots, K\}$. Next, for every obtained partition, W_k dispersion is calculated as presented in Eq. (10), being $C_.$ a group and $d_{..}$ a distance measure. The dispersion representation is shown in Fig. 2(b).

$$D_r = \sum_{i,i' \in C_r} d_{ii'} \tag{9}$$

$$W_k = \sum_{r=1}^{k} \frac{1}{2n_r} D_r \tag{10}$$

Then, the Monte Carlo method is used to generate B reference datasets considering, for example, uniform distributions. Thus, for every reference dataset, the chosen clustering algorithm is performed and their dispersion values W_k^* are estimated. Next, Gap Statistic is calculated by Eq. (11), considering $b = \{1, 2, \ldots, B\}$ and $k = \{1, 2, 3, \ldots, K\}$. Fig. 3(a) demonstrates dispersion values W_ke W_{kb} with the logarithmic function over k clusters.

$$GAP(k) = \left(\frac{1}{B}\right)\sum_{b} \log(W_{kb}^*) - \log(W_k) \tag{11}$$

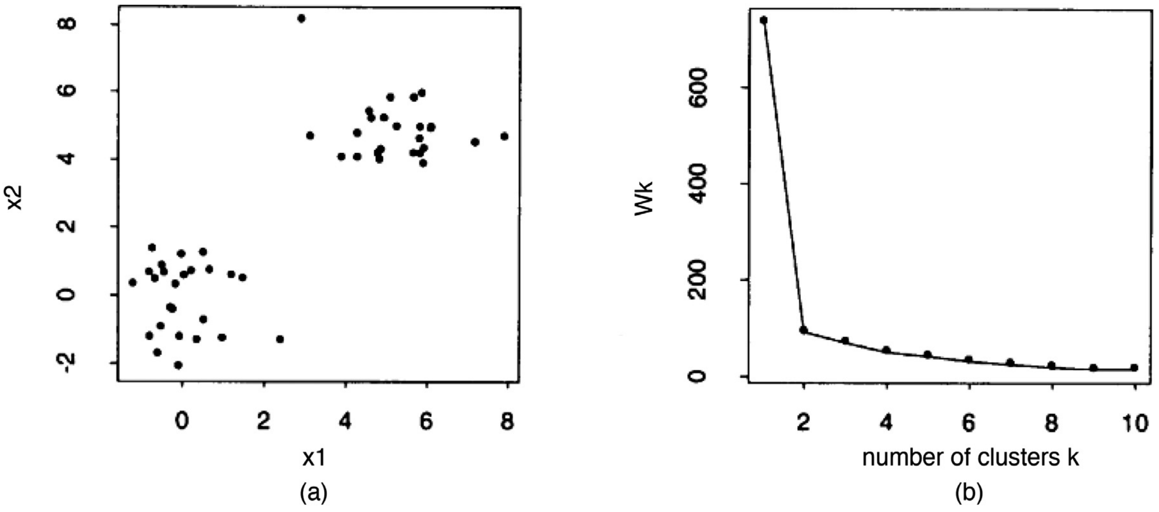


Fig. 2. (a) Dataset representation and (b) W_k dispersion with variation of k number of clusters [14].

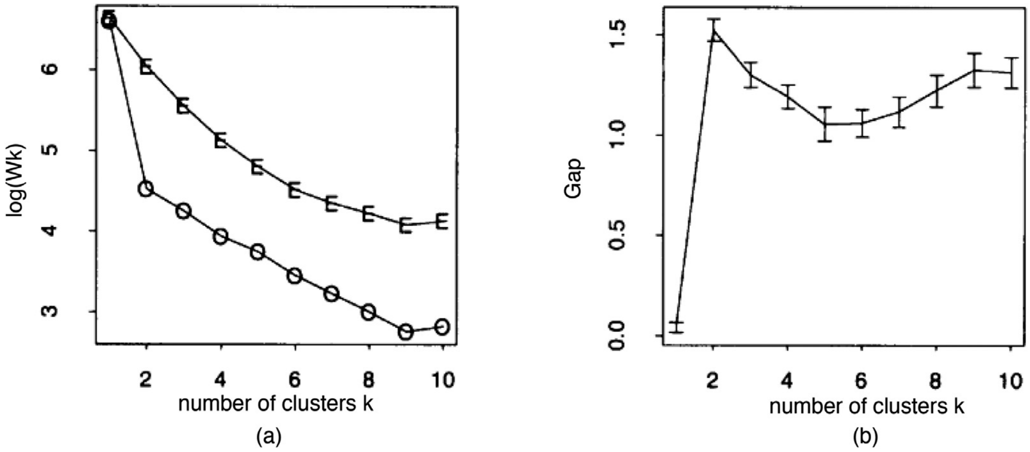


Fig. 3. (a) Function $log(W_k)$(O) and $log(W^*_{kb})$(E) using the Monte Carlo method and (b) the resultant Gap curve [14].

Later, we need to calculated the standard deviation s_k as shown in Eq. (14), which uses Eqs. (13) and (12) to support it.

$$\bar{l} = \left(\frac{1}{B}\right) \sum_b \log(W^*_{kb}) \tag{12}$$

$$sd_k = \left[\left(\frac{1}{B}\right) \sum_b \left\{\log(W^*_{kb}) - \bar{l}\right\}^2\right]^{1/2} \tag{13}$$

$$s_k = sd_k \sqrt{\left(1 + \frac{1}{B}\right)} \tag{14}$$

Finally, the best number of clusters is estimated by considering Eq. (15). According to the authors, the optimal estimation for the best number of clusters is defined by the value that maximizes the Gap Statistic, as shown in Fig. 3(b).

$$\hat{k} = smallest\ k\ such\ as\ GAP(k) \geqslant GAP(k+1) - s_{k+1} \tag{15}$$

2.3. Current limitations

After investigating all pipeline usually adopted in Unsupervised Machine Learning, we have noticed a significant effort to deal with data collected in an independent and identically distributed manner. On the other hand, when data is characterized by presenting a temporal dependency, the number of researches is significantly reduced. There are new or adapted methods designed to calculate distances or cluster time series, however, the validation step is still an open problem and a hot topic, specially, nowadays with the advance of the Data Science area.

To overcome this issue, in this work, we are focused on investigating internal indexes to validate hard partitional clustering applied on top of temporal-dependent data. Aiming at better understanding the development of current researches with a similar focus, we performed a search in the literature to find out related work as described in the following section.

3. Related work

The search for related work was driven by a Systematic Literature Review (SLR), which considered the following main research question: “What are the internal criteria used to validate time series clustering?”. In summary, SLR provides guidelines to make a rigorous search of studies related to a specific topic of interest [15]. Aiming at presenting our manuscript in a more concise way and keeping our focus on the obtained results, this section only discusses the most relevant researches retrieved from SLT in the context of our research. For more details about our SLR steps, the reader can see Appendix A.

The first observation derived from the manuscripts retrieved from SLR is the lack of work proposed to validate clustering methods by using internal criteria. The number of researches is even lower when the dataset is composed of time series. Strongly related to our research question, we have found few work and all of them were published between 2001 and 2018.

On the first analyzed paper [16], authors present an exploratory data-driven strategy based on Unsupervised Fuzzy Clustering Analysis (UFCA) applied in fMRI data. In this work, the authors adapted the Fuzzy C-Means (FCM) algorithm to time domain data and presented a new cluster validity index, referred to as SCF, which combines the CS, S, and fuzzy indexes. Such indexes aim to minimize intra-cluster variance and maximize inter-cluster variance. Briefly, the new index measures the compaction, separation, union, and intersection from the obtained clusters. The results show the advantage of using SCF and its effectiveness to validate time domain data.

In [17], clustering algorithms (e.g. Kohonen's self-organizing map, Minimal free energy vector quantizer and "Neural gas" network) are used in different biomedical applications: (i) fMRI data analysis for human brain mapping; (ii) dynamic contrast magnetic resonance imaging for the diagnosis of cerebrovascular disease; and (iii) breast magnetic resonance imaging for targeting suspected lesions of breast cancer patients. To validate the obtained clusters, three indices are adopted: (i) Kim; (ii) Calinski Harabasz (CH); and (iii) intraclass. Despite the experimental study, the authors emphasize that it is not possible to determine the best index to validate time series clustering in the context of biomedical images.

As the aforementioned work, there exists several similar papers such as [18–22], and [23]. Their main focus is the development of new clustering algorithms or the adaptation of existing ones. In order to validate the resultant clustering, the authors use the traditional indexes as, for instance, Kim, Rand, Silhouette, Calinski-Harabasz, Davies-Bouldin, and Xie-Beni. Thus, one may observe the indices used in all those papers are part of the relative criteria which differs from our proposal.

Authors in [24] proposed a new centroid-based approach to cluster electroencephalographic signals (multi-trial EEG). Their approach yields high-quality multi-trial EEG clustering with respect to the intra-cluster compactness as well as the inter-cluster scatter. In general, the algorithm ensures greater intra-cluster compression or greater inter-cluster dispersion, but not necessarily simultaneously. According to their results, the approach provides highest quality and accuracy for multi-trial EEG data clustering when compared to the other 10 time series clustering algorithms. Such conclusion was drawn after analyzing six validation indexes, including three clustering quality measures: (i) intra-cluster compression; (ii) dispersion *inter-cluster*; (iii) integrated ratio; and (iv) the Rand (RI), F-score and Fleiss' kappa (k) as measurements of cluster precision.

By considering the manuscripts returned by the Systematic Literature Review, the only work developed using Gap Statistic is presented in [25]. In such work, the authors propose a time-varying autoregressive process (TVAR) to describe non-stationary time series and, as a consequence, model them as a mixture of multiple stable autoregressive (AR) processes. Aiming at learning the appropriate number of AR filters needed to model time series, the new technique was based on the internal validation index and Gap Statistic. In summary, the authors use a reference curve to measure how much adding a new AR filter improves the model under reference distribution. Thus, the number of filters is chosen by measuring the maximum gap in comparison with the reference curve.

Besides these manuscripts, we have found important researches developed to cluster time series using robust methods as described in [26–30]. Although these manuscripts are focused on soft clustering, it is worth mentioning their contribution to the clustering literature. A related aspect among their contribution and ours is the adoption of DTW to deal with time series. Moreover, the results presented in these manuscripts emphasize the importance of methods and approaches specifically designed to deal with time series.

In general, the manuscripts depicted in this section use different types of indexes to validate structures yielded from clustering algorithms. Usually, they describe the development of a new algorithm and, then, use well-known validity indexes to compare it with others. Based on our search, just few studies propose new validity indexes. Moreover, most studies use external indices when there is some prior knowledge about the data and relative indices to compare the performance among algorithms. We noticed in this work that internal indexes are less explored, especially when the dataset is characterized by temporal dependencies. Although Gap Statistic was considered in [25], its final purpose was not to assess clustering results.

As widely discussed in the literature, many factors might affect the expected clustering as, for example, the optimal number of groups usually needed by several algorithms. In this sense, internal validity indexes allow to estimate this number without requiring any prior knowledge or analysis on the input data. However, the number of internal indexes is considerably lower when compared to the other ones and we were not able to find any implementation specifically proposed to deal with time series. Indeed, we noticed noted researchers presume the temporal relationship existing in the data does not affect the validity index. This gap motivated us to develop a new internal cluster validation index specifically designed to time series as detailed in the following section.

4. Proposed approach: temporal gap statistic

The lack of mechanisms to validate clustering results in temporal datasets has motivated the design of our approach referred to as Temporal Gap Statistic. It is worth emphasizing our approach is based on an important claim related to the nature of the generation rule that defines the time series behavior. If time series observations are produced by only taking into account stochastic influences, traditional methods devoted to analyzing them in time domain can be used to distinguish their different probability distributions. However, in case of being characterized by deterministic influences, even presenting additive or multiplicative stochastic components, the adaptation of Gap Statistic with Dynamical System methods allows to better model nonlinear and chaotic behavior.

Our approach was obtained after performing three modifications in the original Gap Statistic. The first one was the measure used to calculate the distance between pairs of time series, which is required not only by the clustering algorithms but also by the dispersion W_k presented in Eq. (10). As discussed by several authors, measurements based in Minkowski metric tend to yield unsatisfactory results when similar patterns in time series are displaced over time. An alternative measure is Dynamic Time Warping (DTW), which was detailed in Section 2.1.

An important aspect related to DWT is the lack of support to the triangular inequality, as expected by distance metrics [9], thus missing the property $DTW(x_j, x_n) + DTW(x_n, x_k) \geq DTW(x_j, x_k)$, such that x_j,x_n, and x_k are time series. This is specially important as a basic requirement for specific clustering algorithms. Based on our experiments, whose conclusions were also confirmed by Niennattrakul and Ratanamahatana [31], we have noticed this drawback directly affects the execution of the K-means algorithm [32], originally adopted by Gap Statistic. Once this algorithm is based on minimization of variance in every cluster, such triangular inequality issue has led to produce empty clusters, not respecting the first property presented in Section 2 ($C_i \neq \emptyset, \forall i = \{1, \ldots k\}$).

In order to better understand this problem, we need to briefly describe K-means. This algorithm starts selecting k random instances from the dataset, called centroids, which can be either actual instances or new ones randomly created in the dataset feature space. The k value refers to the expected number of groups. Then, distance measures are used to group instances along with the closest centroid. Next, every k centroid is updated by calculating the average among all instances in the same cluster, thus, the new centroid may represent a completely new instance. Once, our data has temporal dependencies and DTW does not assure the triangular inequality, after the update step, instances in a given cluster can be closer to other centroids than their updated one. As a consequence, empty clusters may be produced.

To solve this problem, we present our second modification on Gap Statistic, changing K-means by K-medoids (also referred to as Partition Around Medoids – PAM) [33]. This clustering algorithm is a variation of K-means that replaces the concept of centroids with medoids. Unlike centroid, medoid is always an actual instance chosen to represent a central point in a cluster. In this case, there will be, at least, one real instance by cluster, which can be the own medoid itself. The next challenge was the generation of random values using a given probability distribution. According to the authors' Gap Statistic, when data is iid, uniform distribution can be used to generate random data to calculate the dispersion. In temporal datasets, though, time series can be created from different and unknown behavior. Therefore, we performed our third modification to create random time series, making sure the new random values respect the feature space that comprises the expected dataset behavior. In summary, by only looking at the dataset in time domain, random values are created without considering the real time series feature space, only randomizing observations between their minimum and maximum values.

Our solution is based on Dynamical System tools [34], which transform time series from the time domain to the phase space. Such space was initially studied by Whitney [35], who applied differential manifolds to reconstruct functions into multidimensional spaces. Based on this reconstruction, Whitney proposed his immersion theorem, which states attractors are better understood when time series are unfolded into a high-dimensional space.

According to Whitney's studies, Takens [36] proved his immersion theorem in which a time series $x_j = (x_{j1}, x_{j2}, \ldots, x_{jd}) \in \mathbb{R}_d$ can be reconstructed into a phase space $x_{jn}(m, \tau) = \{x_{jn}, x_{j(n+\tau)}, \ldots, x_{j(n+(m-1)\tau)}\}$, having $n \in \{1, \ldots, d\}, m$ as the embedded dimension, and τ representing the time delay (or delay dimension or separation dimension). The embedded dimension basically defines the number of axes necessary to unfold time series into the phase space. The delay dimension, on the other hand, is important to represent the seasonal behavior of time series, indicating the necessary displacement among past observations.

The estimation of embedded dimension was studied by Kennel et al. [37], who proposed the False Nearest Neighbors (FNN) method designed to analyze the neighborhood for every observation in the phase space. In summary, this method starts calculating the distance among observations considering the embedded dimension equals to one. Then, a new dimension is added and distances are again calculated. If distances increase, observations are considered as false neighbors, *i.e.*, observations that lay close together are separated in higher embedding dimensions, eliminating the *false neighbors* and making evident the need for a higher dimensional reconstruction. If by adding a new dimension, the false neighbor rate is zero, then the total of dimensions is taken as embedded dimension.

Formally, the False Nearest Neighbors method considers an embedded dimension m, in which the rth neighbor close to the observation x_{jn} is defined by $x^{(r)}_{jn}$. The Euclidean distance among x_{jn} and its rth neighbor is presented in Eq. (16). By adding a new dimension, the time series is reconstructed adding coordinate $(m+1)$ to every vector representing an observation x_{jn}, as represented by term $x_{j(n+mT)}$ in Eq. (17). Hence, this method evaluates the distance variation as new dimensions are added according to Eq. (18).

$$R^2_m(n, r) = \sum_{k=0}^{m-1} (x_{j(n+kT)} - x^{(r)}_{j(n+kT)})^2 \tag{16}$$

$$R^2_{m+1}(n, r) = R^2_m(n, r) + (x_{j(n+mT)} - x^{(r)}_{j(n+mT)})^2 \tag{17}$$

$$V_{n,r} = \sqrt{\frac{R^2_{m+1}(n, r) - R^2_m(n, r)}{R^2_m(n, r)}} = \frac{|x_{j(n+mT)} - x^{(n)}_{j(n+mT)}|}{R^2_m(n, r)} \tag{18}$$

According to Kennel et al. [37], if the distance variation $V_{n,r}$ is greater than a threshold R_{tol}, then observations are considered as false neighbors. As discussed in [37], an acceptable value for this threshold is $R_{tol} \geq 10.0$.

In relation to the delay dimension, there exists several methods to estimate it. In this work, we considered the results presented in [38], which used Average Mutual Information (AMI) to estimate this dimension. In summary, such method works analyzing time series in different time delays. Afterwards, a curve is plotted considering the time delays and the first minimum is adopted as the dimension.

After reconstructing a time series into its phase space, temporal relationships are removed and every dimension can be used to generate random values following some probability distribution. Finally, after randomly generating observations in different dimensions, we reconstruct them back in time domain. This process is repeated to produce as many random time series as necessary to generate reference datasets. The remaining steps follow the original Gap Statistic method.

The steps used to reconstruct an unfolded time series into its time domain, thus keeping the temporal dependency, is summarized in Algorithm 1 . Aiming at better explaining this algorithm, we have considered the resultant time series as a vector of observations and its unfolded version as a matrix.

Algorithm 1: Reconstruction of unfolded time series from the phase space to the time domain.

```
Data: $x_{jn}(m, \tau)$ = unfolded time series, m = embedded
      dimension, $\tau$ = delay dimension
Result: $x_j$ = time series
1 /* Calculate the number of rows in $x_{jn}$.                  */
2 nRows ← nrows($x_{jn}$)
3 /* Start including all elements from the first column
     of $x_{jn}$ as the first time series observations.        */
4 $x_j$ ← $x_{jn}$[1:nRows, 1]
5 for $i$ in 2:$m$ do
6     /* For every matrix column, concatenate the previous
         observations with the remaining ones.                  */
7     $x_j$ ← concatenate($x_j$, $x_{jn}$[(nRows − $\tau$ + 1), $i$])
8 end
```

Aiming at better understanding this algorithm, consider the time series TS-2, presented in Fig. 5. The bottom-most table (in blue) shows this time series unfolded using $m = 3$ and $\tau = 2$ for embedded and delay dimensions, respectively. The reconstruction of this matrix to the time series is started by selecting the elements in the first column $(-9.66, -6.99, -4.96, -3.58, -2.66, -2.11)$, as presented in Line 4. Next, we concatenate elements from the second column, starting from the 5th row $(\text{nRows} - \tau + 1 = 6 - 2 + 1 = 5)$. Then, after the concatenation, the time series contains the observations $(-9.66, -6.99, -4.96, -3.58, -2.66, -2.11, -1.84, -1.78)$. The

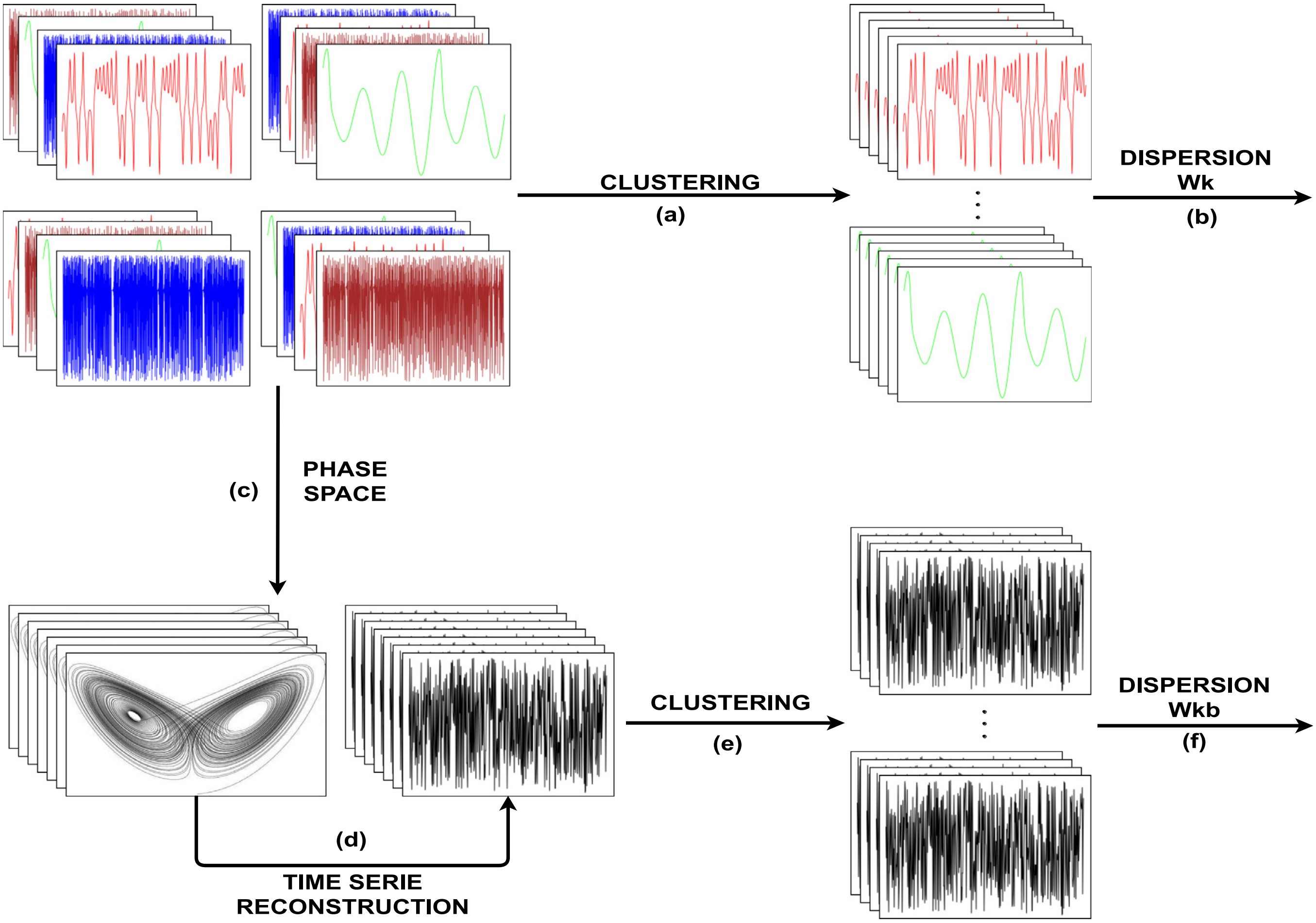
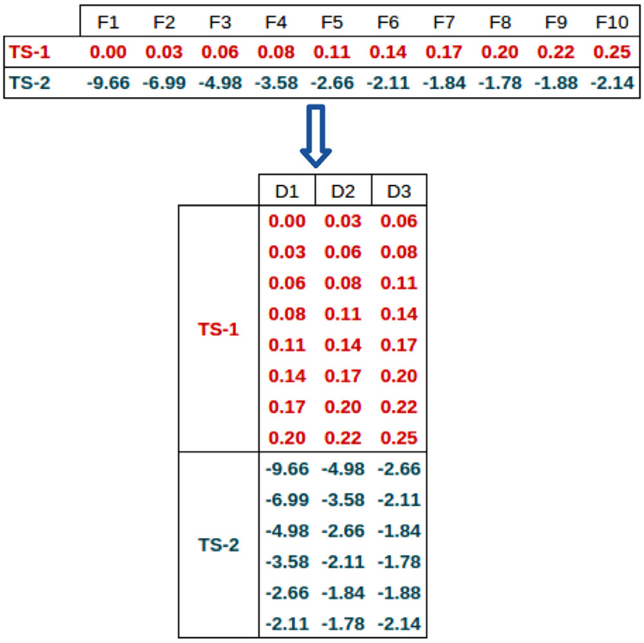


Fig. 4. Temporal Gap Statistic: (a) a dataset composed of time series is initially clustered in the temporal domain; (b) the dispersion W_k is calculated by using Eq. (10); (c) the dataset is unfolded into the phase space; (d) random values are produced by taking into account the spatial limits in the phase space; next, all time series (original and random ones) are reconstructed to the time domain; (e) all time series are clustered; finally, the dispersion W_{kb} is used by the Gap Statistic (Eq. (11)).

same step is performed along with the last column, thus recreating the original time series TS-2.

Fig. 4 summarizes all steps proposed in our Temporal Gap Statistic. Initially, a set of time series, to be clustered, are organized into an feature-value matrix, in which every row is a time series and its observations are organized as columns. Then, in Step (a), a distance matrix is created using DTW, that will be latter used by K-medoid. Then, the dispersion is calculated in Step (b) according to Eq. (10). The clustering and dispersion are run k times and stored in W_k variable, where k represents the number of clusters. Then, our approach transforms all time series into their phase space, as exemplified in Step (c). Next, random values are produced respecting the maximum and minimum values for every dimension. Those random values are reconstructed to the time domain as shown in Step (d). Then, the clustering method is newly executed on the random time series as illustrated in Step (e). The obtained clustering and the dispersion function, Step (f), are executed b times for every k group, whose mean dispersion values is stored in W_{kb}. Finally, given the W_k and W_{kb} dispersion functions, the Gap value for every k group is obtained by Eq. (11).

The most challenging task in our approach is Step (c) which transforms all time series into their phase space to be latter used to yield random observations. As previously mentioned, such transformation uses FNN and AMI, which may estimate different dimensions for each time series. The delay dimension is intrinsically related to the time series and different values will not affect our analysis.



	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
TS-1	0.00	0.03	0.06	0.08	0.11	0.14	0.17	0.20	0.22	0.25
TS-2	-9.66	-6.99	-4.98	-3.58	-2.66	-2.11	-1.84	-1.78	-1.88	-2.14

	D1	D2	D3
TS-1	0.00	0.03	0.06
	0.03	0.06	0.08
	0.06	0.08	0.11
	0.08	0.11	0.14
	0.11	0.14	0.17
	0.14	0.17	0.20
	0.17	0.20	0.22
	0.20	0.22	0.25
TS-2	-9.66	-4.98	-2.66
	-6.99	-3.58	-2.11
	-4.98	-2.66	-1.84
	-3.58	-2.11	-1.78
	-2.66	-1.84	-1.88
	-2.11	-1.78	-2.14

Fig. 5. Example of unfolding two time series (top-most table) into the same embedded dimension (bottom-most table).

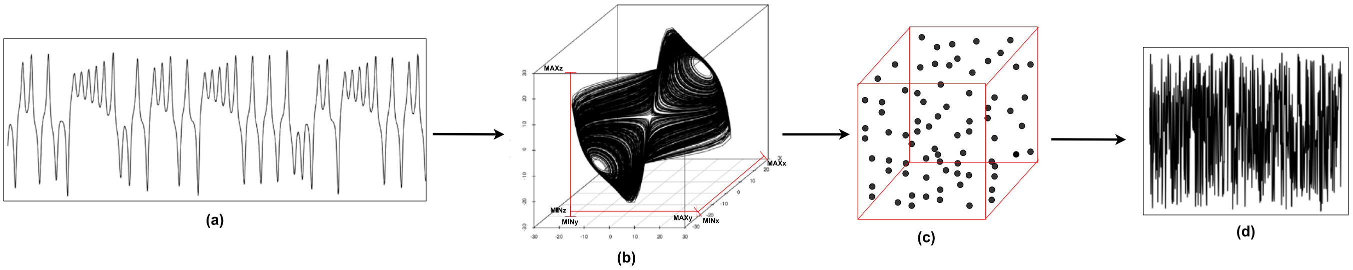


Fig. 6. Process proposed to create random time series: a time series (a) is unfolded into its phase space (b); then, the observation limits are used to create random observations (c) that will be, later, transformed to the time domain (d), thus producing a new random time series.

In relation to the different values for the embedded dimension, our approach was designed based on Whitney's and Takens' researches [35,36], which affirm the adoption of higher embedded dimensions does not affect the time series modeling. For example, if the expected embedded dimension is equal to m, any greater value will produce the same analysis, only demanding more computational time. Aiming at illustrating this step, Fig. 5 shows two time series, TS-1 and TS-2 (top-most table), with 10 observations. Let $m = 2$ and $\tau = 1$ be the estimated embedded and delay dimensions for TS-1. Similarly, the embedded and delay dimensions estimated for TS-2 were $m = 3$ and $\tau = 2$, respectively. Our approach combines all time series into a single data table using the maximum embedded dimension among them ($m = 3$), but respecting every delay dimension as shown in the bottom-most table. Although TS-1 was unfolded with $m = 3$, its original delay dimension ($\tau = 1$) was kept. Therefore, using this bottom-most table, our approach creates a new data table by generating random values within the minimum and maximum values in every dimension (D1, D2, and D3 in our example). The new data table is, then, converted to the time domain (using a inverse version of the unfold process) producing new random time series that truly respect the original time series behavior.

The full process proposed to create random time series is shortly presented in Fig. 6. In this example, we selected a time series produced by the Lorenz system (described in Section 5), whose representation in the time domain is shown in Fig. 6(a). Then, based on its estimated embedded and delay dimensions, such series is unfolded to the phase space as shown in Fig. 6(b). As one may notice, in this example, we used the embedded dimension equals to $m = 3$. By considering the space formed by the 3 dimensions, our approach generates random values as shown in Fig. 6(c). Finally, such random values are transformed to the time domain, producing a new series – Fig. 6(d). This process is repeated inside the Monte Carlo step (see details in Section 2.2.3) to generate several random time series.

5. Experimental setup

In order to assess our Temporal Gap Statistic, we planned some experiments involving four chaotic time series: (i) Lorenz; (ii) Rössler; (iii) Logistic; and (iv) Hénon. The Lorenz time series is produced by system (Eq. (19)) of ordinary differential equations that were initially studied to numerically model some atmospheric phenomena [34,39]. In such system, the variables $\sigma = 10, \beta = 8/3, \rho = 28$ was set to produce a chaotic time series as illustrated in Fig. 7.

$$\begin{cases} \frac{dx}{dt} &= \sigma(y - x) \\ \frac{dy}{dt} &= x(\rho - z) - y \\ \frac{dz}{dt} &= xy - \beta z \end{cases} \tag{19}$$

The Rössler time series is based on Eq. (20), which were defined to model chemical turbulence [34,39]. In such equations, we defined $a = 0.2, b = 0.2, c = 5.7$ to obtain a chaotic time series as well. A sample of this time series is illustrated in Fig. 8.

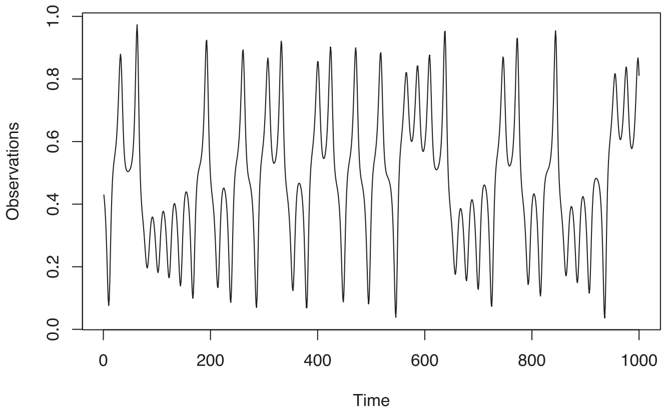


Fig. 7. Chaotic time series created by the Lorenz system.

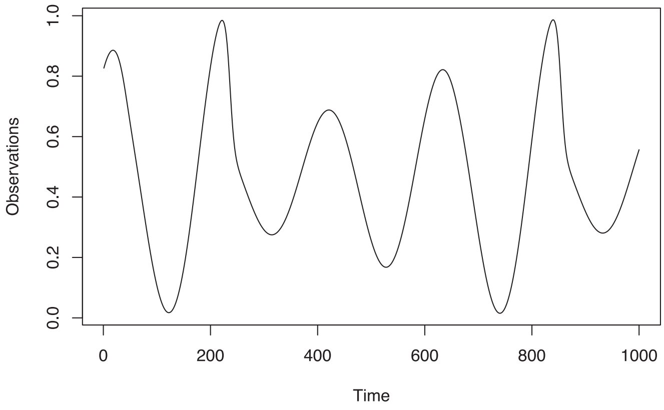


Fig. 8. Chaotic time series created by the Rösler system.

$$\begin{cases} \frac{dx}{dt} &= -x - y \\ \frac{dy}{dt} &= x + ay \\ \frac{dz}{dt} &= b + z(x - c) \end{cases} \tag{20}$$

The Logistic time series is described by the Eq. (21), whose chaotic behavior is obtained using $x_n = 0.5$ and $r = 3.8$ [34,39], as shown in Fig. 9.

$$x_{n+1} = rx_n(1 - x_{n+1}). \tag{21}$$

The last time series used in our experiments is based on Hénon map described in Eq. (22). Although this map exhibits chaotic behavior using different parameters, in our experiments our series were generated with $a = 1.4, b = 0.3$ as exemplified in Fig. 10.

$$\begin{cases} x_{n+1} &= 1 - ax_n^2 + y_n \\ y_{n+1} &= bx_n \end{cases} \tag{22}$$

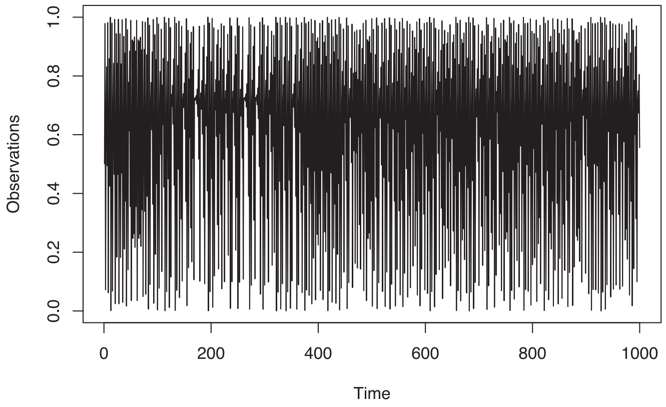


Fig. 9. Chaotic time series created by the Logistic equation.

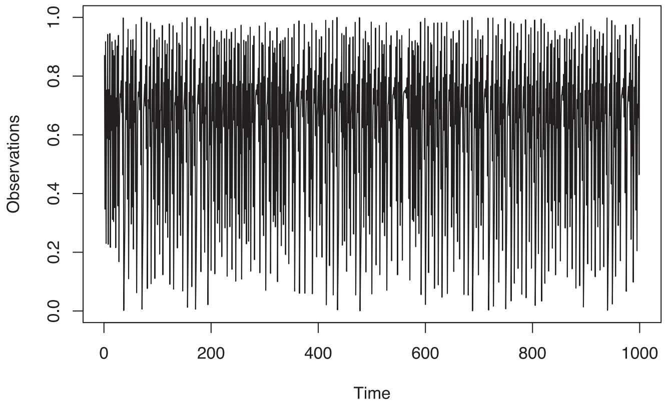


Fig. 10. Chaotic time series created by the Hénon map.

Our experimental datasets were created in 5 steps: i) we created 20,000 observations for every chaotic time series; ii) we normalized all time series within the interval $[0, 1]$; iii) every time series was split in windows of 2,000 observations, resulting in a clean dataset with 40 time series (10 time series, with different observations, for every chaotic behavior); iv) we generated noisy time series following a normal distribution $\mathcal{N}(\mu, \sigma)$ with mean and standard deviation equals to 0 and 0.1; and v) the noisy time series were added to chaotic ones producing a noisy dataset with 4 groups with 10 time series.

The methodology adopted in our experiments was conducted in three phases. In the first one, different pairs of groups were used to test our approach. The second phase was performed using different combinations of three groups and, finally, all four groups were evaluated.

6. Results

The effectiveness of our Temporal Gap Statistic approach as a new internal validity index was evaluated to determine the optimal number of clusters using the datasets previously presented.

The first set of experiments was conducted on the clean dataset as shown in Figs. 11–16. The results were obtained after combining all time series to form 2, 3, and 4 clusters. In every figure, we show two plots to represent the outputs produced by our approach. The first one presents the W_k and W_{kb} dispersion values along with the number of k clusters. The second plot illustrates the Gap values and the number of k clusters.

In our experiments, we tested the cluster numbers within the interval $k = [1, 10]$. The best number of groups is given when W_k is lower than reference curve W_{kb}, corresponding to the maximization of the gap value. For example, by looking at Fig. 11, we notice a sudden reduction of W_K when $k = 2$. The resulting analysis of this process can be observed on the right-most plot in which the Gap curve is displayed with standard error bars produced by Eq. (15). Thus, the Gap curve has a clear maximum at $k = 2$, suggesting the estimated number of clusters is 2, emphasizing there exists 2 groups of time series produced by the Lorenz and Hénon equations. Similar conclusion is drawn when we analyze the pairs of groups produced by other equations (including Rössler and Logistic) as shown in Figs. 12 and 13.

Still considering this clean dataset, we tested our approach in a scenario with 3 groups as shown in Figs. 14 and 15. Both figures show a large difference between W_k and W_{kb}, providing a margin significantly large when $k = 2$. However, we also notice this margin still increases when we verify a higher number of clusters. In order to better support our analysis, we analyze the Gap curves (right-most plots) that show the maximum gap in $k = 3$. Therefore, the estimated numbers of clusters in both experiments are equal to the expected ones.

In our last experiment with the clean dataset, we used the four groups of time series. The obtained results were similar to those ones previously obtained, emphasizing our method has correctly estimated the expected number of groups ($k = 4$) as shown in Fig. 16.

Our second set of experiments, we analyzed the performance of our approach on the noisy dataset. This experiment was specially important to assess the influence of noise on the Temporal Gap Statistic. For this reason, we considered time series, whose observations were affected by about 10% of noise. We followed the same experimental process considered in previous analyses. According to the results presented in Figs. 17–22, one may notice our approach

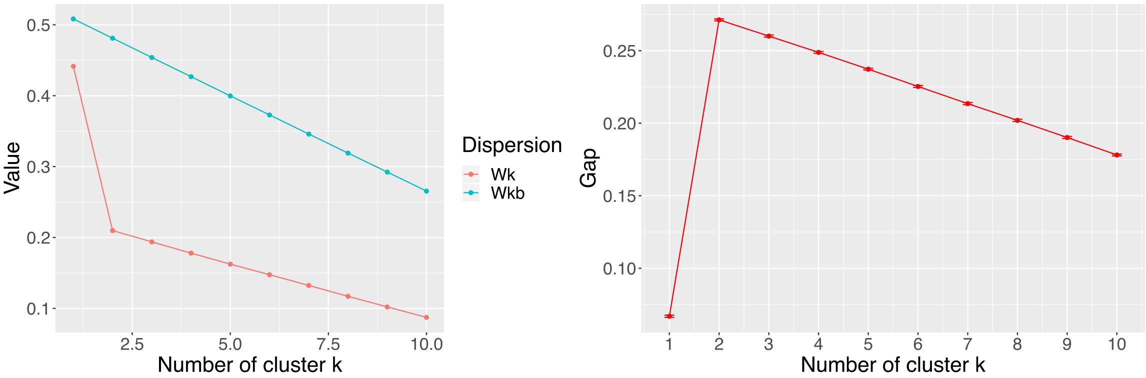


Fig. 11. Experiment using two groups of time series from the clean dataset, produced by the Lorenz and Hénon equations.

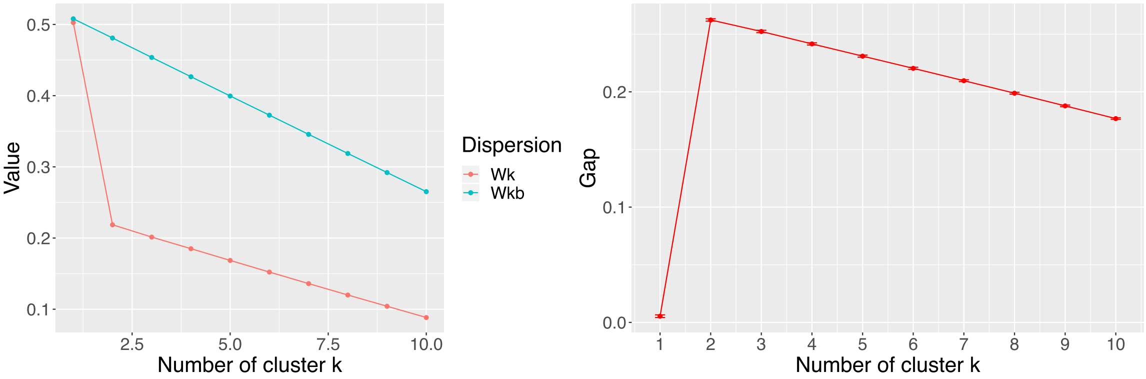


Fig. 12. Experiment using two groups of time series from the clean dataset, produced by the Lorenz and Logistic equations.

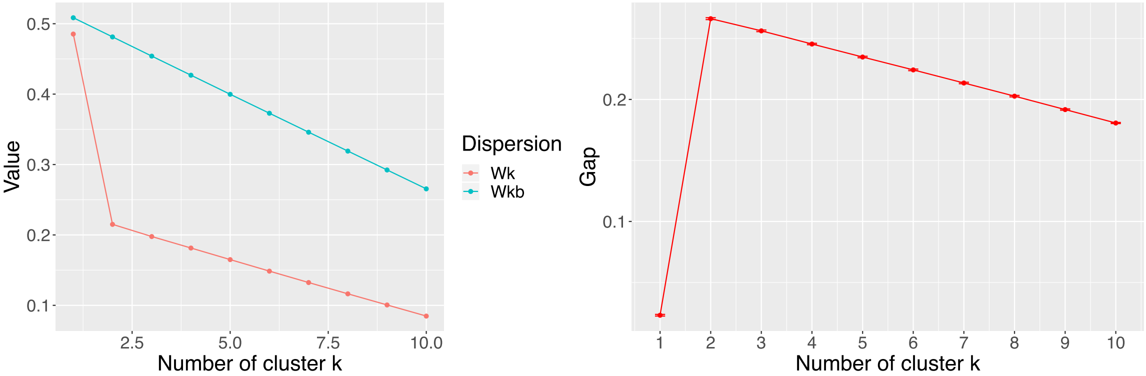


Fig. 13. Experiment using two groups of time series from the clean dataset, produced by the Rössler and Logistic equations.

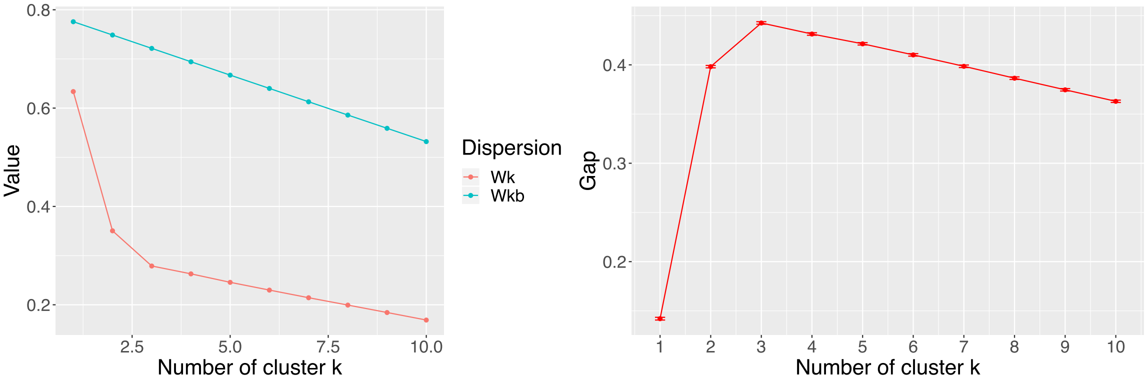


Fig. 14. Experiment using three groups of time series from the clean dataset, produced by the Lorenz, Rössler, and Logistic equations.

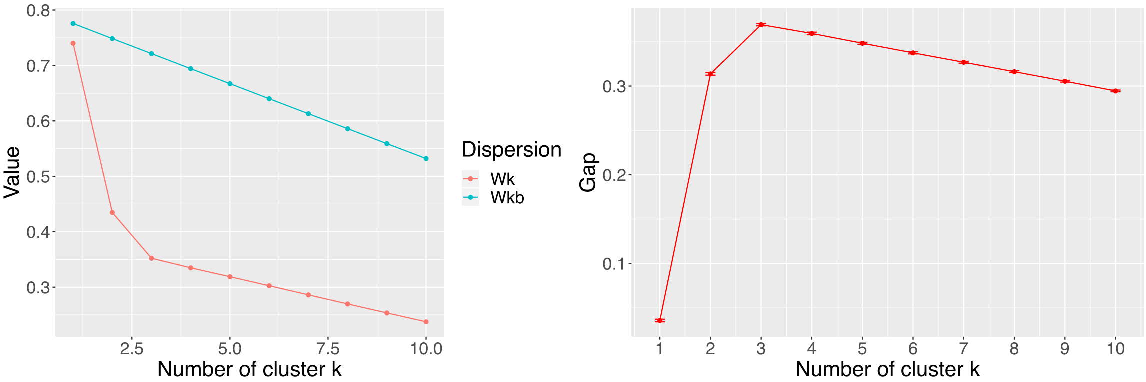


Fig. 15. Experiment using three groups of time series from the clean dataset, produced by the Rössler, Logistic, and Hénon equations.

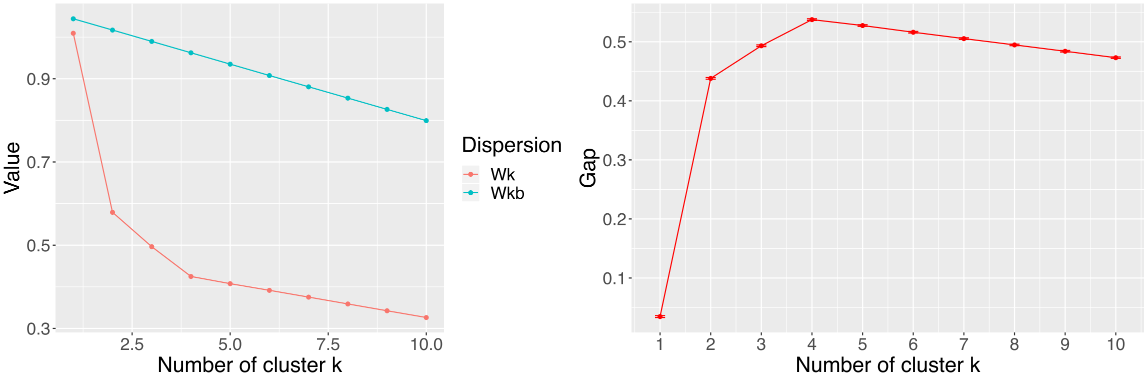


Fig. 16. Experiment using four groups of time series from the clean dataset, produced by the Lorenz, Rössler, Logistic, and Hénon equations.

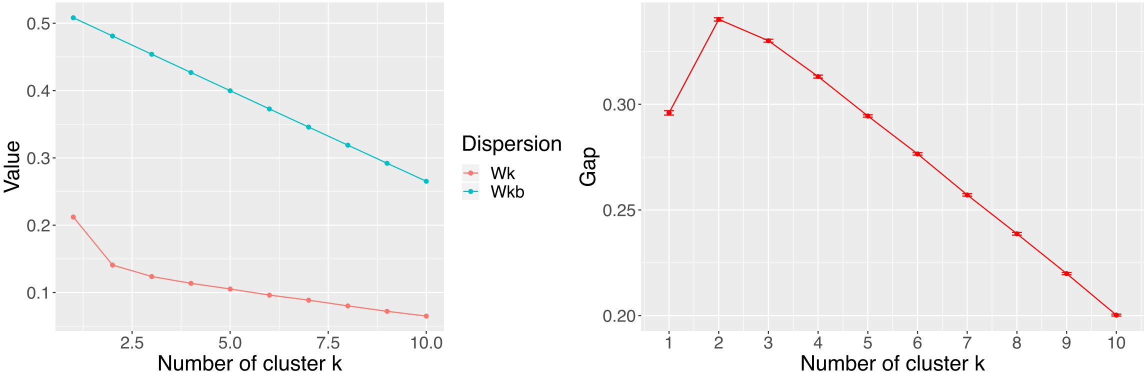


Fig. 17. Experiment using two groups of time series from the noisy dataset, produced by the Lorenz and Rössler equations.

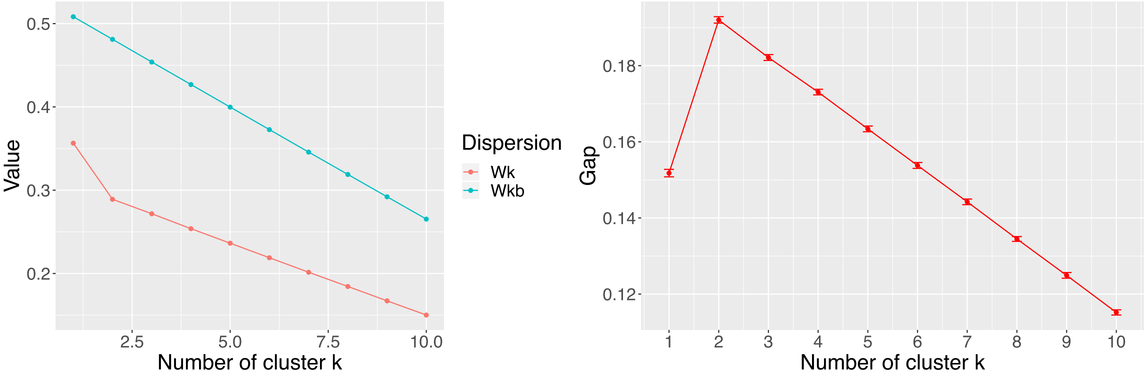


Fig. 18. Experiment using two groups of time series from the noisy dataset, produced by the Logistic and Hénon equations.

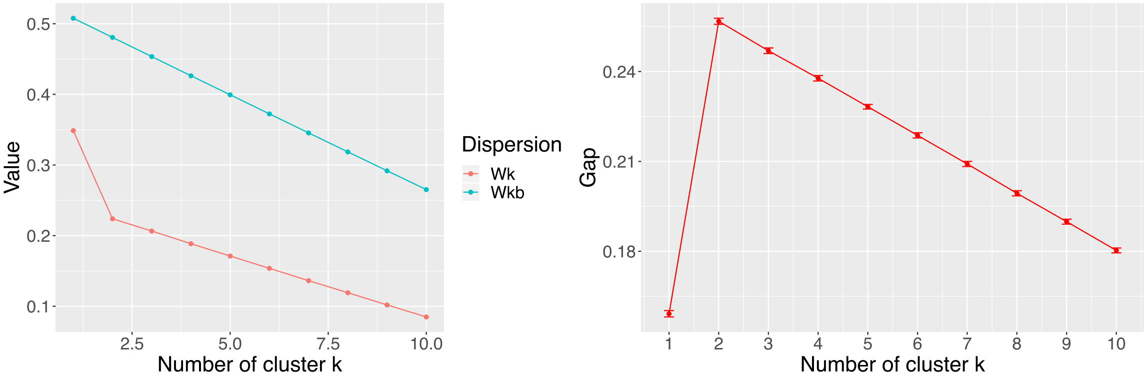


Fig. 19. Experiment using two groups of time series from the noisy dataset, produced by the Rössler and Logistic equations.

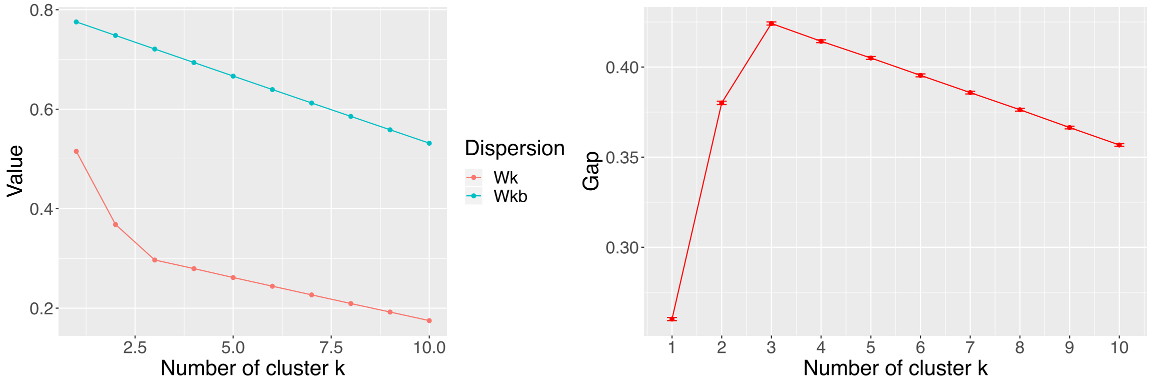


Fig. 20. Experiment using three groups of time series from the noisy dataset, produced by the Lorenz, Rössler, and Logistic equations.

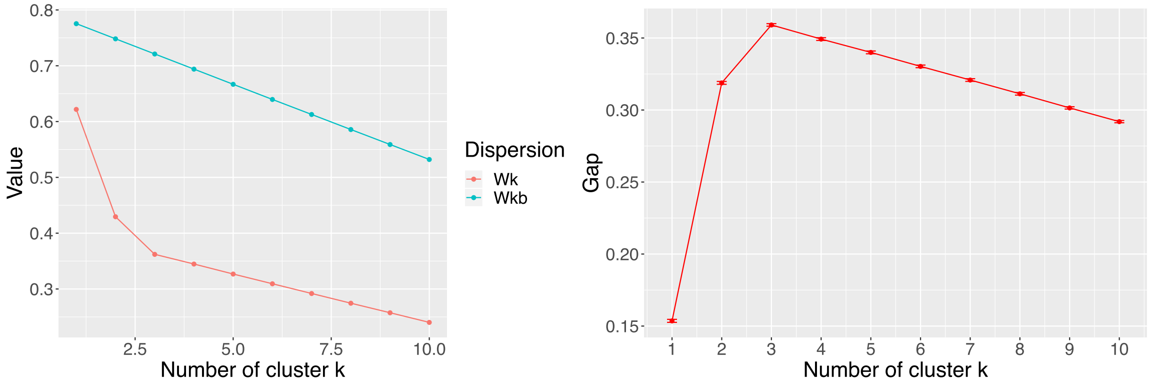


Fig. 21. Experiment using three groups of time series from the noisy dataset, produced by the Lorenz, Logistic, and Hénon equations.

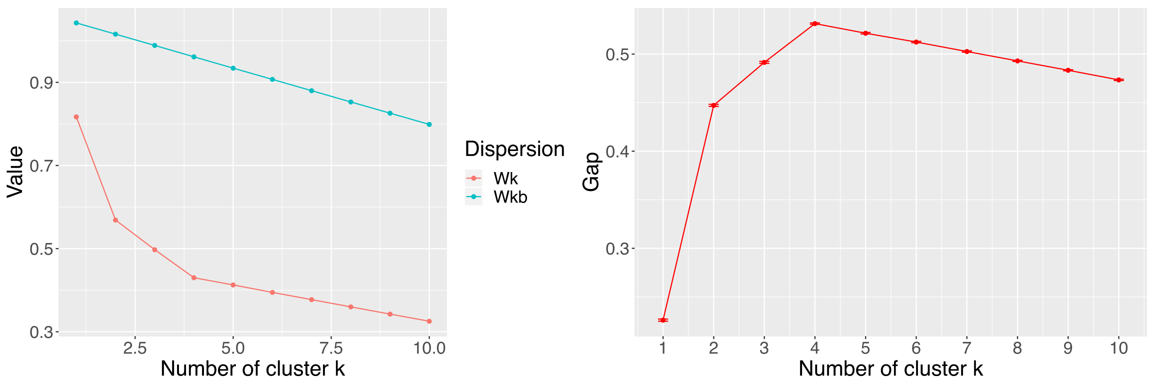


Fig. 22. Experiment using four groups of time series from the noisy dataset, produced by the Lorenz, Rössler, Logistic and Hénon equations.

presented similar results, as obtained on the clean dataset, thus accurately estimating the number of clusters.

Finally, we also evaluated our approach using standard external indices, which are usually adopted when the expected partition is previously known. After selecting the optimal number of clusters using our approach, the final partition (clusters) was calculated using the K-medoid algorithm. We selected the Rand, Jaccard, Folkes-Mallows, and Γ indices discussed in Section 2.2.1. We used those indices to make sure our approach was not only selecting the best number of groups, but also accordingly clustering the time series. All results obtained by the indices were equal to 1, emphasizing we have found the best structure using our approach.

7. Conclusions

This paper presented a new internal index to validate partition produced by clustering algorithms on temporal datasets. This new index was designed on top of Gap Statistic after specifically modifying three main parts. The first one was the adoption of DTW to measure distances between dataset instances, which are time series in our context. This modification was important to take into account displacements between time series during no only the clustering execution but also to calculate the dispersion W_k. Our second modification was the adoption of a clustering algorithm based on medoids instead of centroids, aiming at avoiding to fail the cluster restriction on having empty clusters. Finally, our deepest modification was in the Monte Carlo method used to generate reference datasets with random values. Rather than generate such values in time domain, we transform all time series into the phase space using Dynamical System tools.

As pointed out by our Systematic Literature Review, the reduced number of internal indexes devoted to validating clustering algorithms and the lack of such indexes to deal with temporal data have motivated this work. In this sense, our experiments provided outstanding results, thus emphasizing the importance of our approach to estimate the optimal number of groups from temporal dataset, which is an important task in Unsupervised Machine Learning. As proof of concept, our experiments were conducted using synthetic datasets, similarly as the original Gap statistic manuscript [14]. Aiming at simulating a real-world scenario, we have considered different influences of noise as well as different parts of time series created by chaotic processes. By using such modifications, we were able to change the time series observations without modifying the expected behavior. Thus, our experiments were executed by exploring different situations, regardless of the generation process being synthetic or real.

Another important discussion is related to the number of observations. Although we have analyzed time series with the same length, this is not a requirement to run our approach for two reasons. Firstly, the DTW distance can be calculated between time series with different sizes, as discussed in [9]. Secondly, by combining unfolded time series, the different number of rows in every resulting matrix (Fig. 5) is just used to define the spatial limits, as shown in Fig. 6(c), where the random points will be placed at. Furthermore, it is important to highlight that very short time series can also be analyzed by our approach, once none of the adopted methods has such a limitation (e.g. the time series used in Fig. 5 presented only 10 observations), however, the final results might be affected whether the attractor is not described. As a future work, we intend to assess our approach to find out structures, as motifs, in a single time series. We believe this analysis might be useful to, for example, detect concept drift. "

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Systematic Literature Review

This section presents the our Systematic Literature Review, which were divided in three phases: i) Phase I, where some criteria were defined to seek related work; ii) Phase II analyzes and quantifies the quality of collected papers, and iii) Phase III presents conclusions obtained from the systematic study.

A1. Phase I

In this first phase, we defined the general scope of the research by defining the research criteria to characterize whether a paper is related to our study or not. In this sense, we defined the objective of this research, the main and secondary research questions, the search repositories, the standard language, the list of keywords, the search query, the inclusion and exclusion criteria, and, finally, the general process of execution.

As previously discussed, the main objective of this research is to find out internal criteria used in cluster validity for data with temporal dependence. Based on this objective, the main question to guide this research is:

$$What\ are\ the\ internal\ criteria\ used\ in\ time\ series\ cluster\ validity?$$

In addition to this main question, it is relevant to define a set of secondary questions, which are directly associated to the validity of the proposed research. This set of questions aims at discovering practical applications of the research, evaluating techniques, and understanding publication trends. The secondary questions are:

SQ.1 - What types of practical applications clustering validity for time series can be used for?
SQ.2 - How is clustering validity used?
SQ.3 - Why should we use clustering validity for time series?
SQ.4 - What are the main indexes used in time series cluster validity?
SQ.5 - What is the frequency of published papers per year?
QS.6 - What are the limitations of these indexes?

After defining these questions, the next step of this phase was to choose the search repositories, from which related studies were obtained. We selected repositories which provided web search engines that accept queries using keywords and are commonly used by the scientific community. Based on these restrictions, the following repositories were chosen:

- Scopus (https://www.scopus.com/)
- ACM Digital Library (https://dl.acm.org/)
- IEEE Xplore Digital Library (https://ieeexplore.ieee.org/)

The standard language used in this systematic review was English. As the next step, the following keywords were chosen considering the hypothesis and the main question of this review:

- Data Organization: Time Series
- Goals: Internal Criteria
- Results: Cluster Validity

Based on these keywords, the following search query was defined:

("time series") AND ("internal criteria") AND ("cluster validity")

Due to the fact that no relevant papers to the study were found in the repositories using this query, a new search with a general context was elaborated:

("time series") AND ("cluster validity")

In addition to the papers returned with the previous words, we also decided to search for manuscripts that specifically use Gap Statistic to time series cluster validity. To do so, the following search query was defined:

(”time series") AND (”gap statistic")

Aiming at selecting relevant papers for this systematic review, an evaluation was performed to define which one would be included or not in the review. In this filtering, we chose to include works that clearly define the internal, external or relative criteria used in time series cluster validity. However, the exclusion of papers was performed whenever the papers did not present a satisfactory cluster validity process. In addition, papers were discarded when they did not make a clear presentation about validity use or when they present a redundant context.

Therefore, Phase I presents the initial conditions of papers selection by systematic review. The next phase is the analysis of papers selected in this first phase.

A2. Phase II

Through application of keywords on the selected repositories, it was returned 54 papers, where its distribution can be viewed in Table A.1. After, the inclusion and exclusion was performed by the reading of their titles and abstracts. As result, the majority of papers were excluded to present redundant content or do not present a detail research on the context of validity index and time series. Therefore, it was classified 10 papers as strongly related to the subject presented in this work.

Table A1
Numbers of papers find through Systematic Review.

Repository	Number of papers
ACM	1
IEEE	13
Scopus	40
Total	**54**
Inclusion	**10**
Exclusion	**44**

Table A2
Number of papers published by year.

Year	Frequency
2001	1
2004	1
2007	1
2011	1
2015	1
2016	2
2017	2
2018	1

Aiming to answer secondary question SQ.5, it was analyzed the frequency of papers published per year. Despite the low amount of papers, it is possible to note that they are recent publications are recently, from 2001 to 2018.

The most related manuscripts were detailed in Section 3, thus answering SQ.2. The answer for Question SQ.4, the most used indexes, is: Dunn, Calisnski Harabasz, Silhuette, Rand, Davies-Bouldin, Weinmert Gancarski, PBM, homogeneity, weighted, inter-intra, Krzanowski-Lai, Xie-Beni, Intraclass, Kim, R, SCF, Variation of Information, Normalized Mutual Information, Gap Statistic.

Finally, it is important to highlight that such indexes are applied to the most different types of applications (SQ.1), demonstrating their relevance when analyzing clusters in temporal data (SQ.3).

A3. Phase III: final considerations

In general, the reported works use different types of indexes to cluster validity: external, relative or internal criteria. In order to obtain consistency in the results, it is observed that several indexes are used to evaluate the clustering. In this case, the best partitioning is chosen based on the execution that provided the highest indexes.

Table A.2 shows the number of papers published by year according to our SLR. Based on this table, we notice this is a recent research subject. Moreover, the number of manuscripts shows time series clustering is still an open problem specially because Data Stream Analysis and Concept Drift methods are considered hot topics in Data Science area.

CRediT authorship contribution statement

Rosana Guimarães Ribeiro: Methodology, Software, Validation, Writing - original draft. **Ricardo Rios:** Conceptualization, Methodology, Writing - review & editing, Supervision.

References

[1] Mitchell TM, et al. Machine learning, 45. Burr Ridge, IL: McGraw Hill; 1997. p. 870–7.
[2] Bishop CM. Pattern recognition and machine learning (information science and statistics). Secaucus, NJ, USA: Springer-Verlag New York, Inc; 2006.
[3] Faceli K, Lorena AC, Gama J, Carvalho ACPF. Inteligência artificial: uma abordagem de aprendizado de máquina. LTC; 2011.
[4] Jain AK, Dubes RC, et al. Algorithms for clustering data, 6. Prentice Hall Englewood Cliffs; 1988.
[5] Xu R, Wunsch D. Clustering, 10. John Wiley & Sons; 2008.
[6] Theodoridis S, Koutroumbas K. Clustering: basic concepts. Pattern Recognit. 2006:483–516.
[7] Berndt DJ, Clifford J. Using dynamic time warping to find patterns in time series.. In: KDD Workshop, 10. Seattle, WA; 1994. p. 359–70.
[8] Vapnik VN. An overview of statistical learning theory. Trans Neur Netw 1999;10(5):988–99. doi:10.1109/72.788640.
[9] Ding H, Trajcevski G, Scheuermann P, Wang X, Keogh E. Querying and mining of time series data: experimental comparison of representations and distance measures. Proc VLDB Endow 2008;1(2):1542–52. doi:10.14778/1454159.1454226.
[10] Tormene P, Giorgino T, Quaglini S, Stefanelli M. Matching incomplete time series with dynamic time warping: an algorithm and an application to post-stroke rehabilitation. Artif Intell Med 2009;45(1):11–34. doi:10.1016/j.artmed.2008.11.007.
[11] Duarte FS, Rios RA, Hruschka ER, de Mello RF. Decomposing time series into deterministic and stochastic influences: a survey. Digit Signal Process 2019:102582.
[12] Aghabozorgi S, Shirkhorshidi AS, Wah TY. Time-series clustering–a decade review. Inf Syst 2015;53:16–38.
[13] Vendramin L, Campello RJ, Hruschka ER. On the comparison of relative clustering validity criteria. In: Proceedings of the 2009 SIAM international conference on data mining. SIAM; 2009. p. 733–44.
[14] Tibshirani R, Walther G, Hastie T. Estimating the number of clusters in a data set via the gap statistic. J R Stat Soc Series B (Statistical Methodology) 2001;63(2):411–23.
[15] Kitchenham B, Brereton OP, Budgen D, Turner M, Bailey J, Linkman S. Systematic literature reviews in software engineering - a systematic literature review. Information and Software Technology 2009;51(1):7–15. doi:10.1016/j.infsof.2008.09.009. Special Section - Most Cited Articles in 2002 and Regular Research Papers http://www.sciencedirect.com/science/article/B6V0B-4TX182T-2/2/d714d8469c560c40f3cdb6bce5534036.
[16] Fadili M-J, Ruan S, Bloyet D, Mazoyer B. On the number of clusters and the fuzziness index for unsupervised FCA application to bold fMRI time series. Med Image Anal 2001;5(1):55–67.
[17] Meyer-Bäse A, Saalbach A, Lange O, Wismüller A. Unsupervised clustering of fMRI and MRI time series. Biomed Signal Process Control 2007;2(4):295–310.
[18] Himberg J, Hyvärinen A, Esposito F. Validating the independent components of neuroimaging time series via clustering and visualization. Neuroimage 2004;22(3):1214–22.
[19] Maji P, Paul S. Microarray time-series data clustering using rough-fuzzy c-means algorithm. In: 2011 IEEE International conference on bioinformatics and biomedicine. IEEE; 2011. p. 269–72.
[20] Salgado CM, Ferreira MC, Vieira SM. Mixed fuzzy clustering for misaligned time series. IEEE Trans Fuzzy Syst 2017;25(6):1777–94.
[21] Das SP, Padhy S. Unsupervised extreme learning machine and support vector regression hybrid model for predicting energy commodity futures index. Memetic Comput 2017;9(4):333–46.
[22] Homenda W, Jastrzebska A. Clustering techniques for fuzzy cognitive map design for time series modeling. Neurocomputing 2017;232:3–15.
[23] Fahiman F, Bezdek JC, Erfani SM, Palaniswami M, Leckie C. Fuzzy c-Shape: a new algorithm for clustering finite time series waveforms. In: 2017 IEEE International conference on fuzzy systems (FUZZ-IEEE). IEEE; 2017. p. 1–8.
[24] Dai C, Pi D, Cui L, Zhu Y. MTEEGC: A novel approach for multi-trial eeg clustering. Appl Soft Comput 2018;71:255–67.
[25] Ding J, Noshad M, Tarokh V. Learning the number of autoregressive mixtures in time series using the gap statistics. In: 2015 IEEE International conference on data mining workshop (ICDMW). IEEE; 2015. p. 1441–6.
[26] D'Urso P, De Giovanni L, Massari R. Trimmed fuzzy clustering of financial time series based on dynamic time warping. Ann Oper Res 2019:1–17.
[27] Lafuente-Rego B, D'Urso P, Vilar J. Robust fuzzy clustering based on quantile autocovariances. Stat Pap 2018:1–56.
[28] D'Urso P, De Giovanni L, Massari R. Robust fuzzy clustering of multivariate time trajectories. Int J Approx Reason 2018;99:12–38.
[29] Vilar JA, Lafuente-Rego B, D'Urso P. Quantile autocovariances: a powerful tool for hard and soft partitional clustering of time series. Fuzzy Sets Syst 2018;340:38–72.
[30] D'Urso P, De Giovanni L, Massari R. GARCH-based robust clustering of time series. Fuzzy Sets Syst 2016;305:1–28.
[31] Niennattrakul V, Ratanamahatana CA. On clustering multimedia time series data using k-means and dynamic time warping. In: 2007 International conference on multimedia and ubiquitous engineering (MUE'07). IEEE; 2007. p. 733–8.
[32] Lloyd S. Least squares quantization in PCM. IEEE Trans Inf Theory 1982;28(2):129–37. doi:10.1109/TIT.1982.1056489.
[33] Kaufman L, Rousseeuw PJ. Partitioning around medoids (program PAM). In: Finding groups in data: an introduction to cluster analysis; 1990. p. 68–125.
[34] Alligood K, Sauer T, Yorke J. Chaos: an introduction to dynamical systems. Textbooks in mathematical sciences. Springer New York; 1997.
[35] Whitney H. Differentiable manifolds. Ann Math 1936;37(3):645–80.

[36] Takens F. Detecting strange attractors in turbulence. In: Dynamical systems and turbulence, Warwick 1980. Springer; 1981. p. 366–81.

[37] Kennel MB, Brown R, Abarbanel HDI. Determining embedding dimension for phase-space reconstruction using a geometrical construction. Phys Rev A 1992;45(6):3403–11. doi:10.1103/PhysRevA.45.3403.

[38] Fraser AM, Swinney HL. Independent coordinates for strange attractors from mutual information. Phys Rev A 1986;33(2):1134–40. doi:10.1103/PhysRevA.33.1134.

[39] Swiercz E. A new method of detection of coded signals in additive chaos on the example of barker code. Signal Process 2006;86(1):153–70.