Pattern Recognition Letters 75 (2016) 63–69

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

Generalized *k*-means-based clustering for temporal data under weighted and kernel time warp☆

Saeid Soheily-Khah, Ahlame Douzal-Chouakria*, Eric Gaussier

Universite Grenoble Alpes, CNRS-LIG/AMA, France

ARTICLE INFO

Article history:
Received 8 June 2015
Available online 18 March 2016

Keywords:
Temporal data
Time series
Clustering
Time warp
Averaging time series
Temporal alignment kernel
Generalized *k*-means
Kernel *k*-means

ABSTRACT

Temporal data naturally arise in various emerging applications, such as sensor networks, human mobility or internet of things. Clustering is an important task, usually applied *a priori* to pattern analysis tasks, for summarization, group and prototype extraction; it is all the more crucial for dimensionality reduction in a big data context. Clustering temporal data under time warp measures is challenging because it requires aligning multiple temporal data simultaneously. To circumvent this problem, costly *k*-medoids and kernel *k*-means algorithms are generally used. This work investigates a different approach to temporal data clustering through weighted and kernel time warp measures and a tractable and fast estimation of the representative of the clusters that captures both global and local temporal features. A wide range of 20 public and challenging datasets, encompassing images, traces and ECG data that are non-isotropic (*i.e.*, non-spherical), not well-isolated and linearly non-separable, is used to evaluate the efficiency of the proposed temporal data clustering. The results of this comparison illustrate the benefits of the method proposed, which outperforms the baselines on all datasets. A deep analysis is conducted to study the impact of the data specifications on the effectiveness of the studied clustering methods.

1. Introduction and related work

Temporal data naturally arise in various emerging applications, such as sensor networks, human mobility or internet of things. Clustering is an important task, usually applied *priori* to any pattern analysis tasks, for summarization, cluster and prototype extraction, and is crucial for big data dimensionality reduction.

k-means-based clustering, viz. standard *k*-means, *k*-means++, fuzzy *c*-means, and all its variations, is among the most popular clustering algorithms, because it provides a good trade-off between the quality of the solution obtained and its computational complexity [1]. However, the *k*-means clustering of temporal data under the commonly used dynamic time warping (DTW) or the several well-established temporal kernels [2–6] is challenging because estimating cluster centroids requires aligning multiple temporal data simultaneously. Such alignments, referred to as multiple sequence alignments [7,8], become computationally prohibitive and impractical when the data size increases. Some progressive and iterative heuristics have been proposed but are limited to the standard DTW and to temporal data of the same global behavior [8–11]. For temporal data clustering, to bypass the centroid estimation problem, *k*-medoids and kernel *k*-means [12,13] are generally used [14]. For *k*-medoids, a medoid is a good representative of data that have the same global dynamic but inappropriate for capturing local temporal features [15]. For kernel *k*-means, although efficient for non-linearly separable clusters because centroids cannot be estimated in the Hilbert space, it refers to pairwise comparisons for the cluster assignment step. While *k*-means-based clustering, of linear complexity, remains a fast algorithm, *k*-medoids and kernel *k*-means have a quadratic complexity due to the pairwise comparisons involved. This work proposes a fast and accurate approach that generalizes the *k*-means-based clustering algorithm for temporal data based on i) an extension of the standard time warp measures to consider both global and local temporal differences and ii) a tractable and fast estimation of the cluster representatives based on the extended time warp measures. Temporal data centroid estimation is formalized as a non-convex quadratic constrained optimization problem. The developed solutions allow for estimating not only the temporal data centroid but also its weighting vector, which indicates the representativeness of the centroid elements. The solutions are particularly studied under the standard dynamic time warping [5], dynamic temporal alignment kernel [6] and global alignment kernels [3,4]. A wide range of 20 public and challenging datasets, which are non-isotropic (*i.e.*, non-spherical) and not well-isolated (and thus non-linearly separable), is used to compare the proposed approach with *k*-medoids

☆ This paper has been recommended for acceptance by G. Moser.

* Corresponding author: Tel.: +33 457421485.
E-mail address: Ahlame.Douzal@imag.fr (A. Douzal-Chouakria).

http://dx.doi.org/10.1016/j.patrec.2016.03.007
0167-8655/

and kernel k-means. The results of this comparison illustrate the benefits of the method proposed, which outperforms the other methods on all datasets. The impact of the isotropy and isolation of clusters on the effectiveness of the clustering methods is also discussed. In Section 2, we present the generalized k-means-based clustering for temporal data, the proposed solutions being directly applicable to any other variations of the k-means algorithm. We then give an extension of the standard time warp measures and discuss their properties. In Section 3, we describe the centroid estimation procedure based on the extended weighted and kernel time warp measures. The conducted experimentations and results obtained are discussed in Section 4. The main contributions of the paper are as follows:

1. We propose a generalization of the k-means-based clustering to temporal data under time warp measures.
2. We extend dynamic time warping and temporal alignment kernels to capture both global and local differences.
3. We propose a fast and tractable estimation of the cluster representatives under the extended time warp measures.
4. We show, through a deep analysis of a wide range of 20 non-isotropic, linearly non-separable public data, that the proposed solutions are faster and outperform the alternative methods.

In the remainder of the paper, we use bold, lower-case letters for vectors, time series and alignments, the context being clear to differentiate between these elements.

2. Generalized k-means for temporal data clustering

The k-means algorithm aims at providing a partition of a set of data points in distinct clusters such that the inertia within each cluster is minimized, the inertia being defined as the sum of distances between any data point in the cluster and the centroid (or representative) of the cluster. The k-means algorithm was originally developed with the Euclidean distance, the representative of each cluster being defined as the center of gravity of the cluster. This algorithm can, however, be generalized to arbitrary dissimilarities (resp. similarities) by replacing the centroid update step with an explicit optimization problem that yields, for each cluster, the representative that minimizes (resp. maximizes) the total dissimilarity (resp. similarity) to all data points of that cluster [16–18]. We focus in this study on (dis)similarity measures between time series that are based on time alignments because such measures (which encompass the standard dynamic time warping and its variants/extensions) are the ones most commonly used in the literature. We first define below what an alignment between two time series is *prior* to introducing the generalized k-means algorithm for temporal data under extended time warp measures.

In the remainder of the paper, X denotes a set of univariate discrete time series $\mathbf{x} = (x_1, \ldots, x_T)$ of assumed length T.[1] An alignment $\boldsymbol{\pi}$ of length $|\boldsymbol{\pi}| = m$ between two time series $\mathbf{x}_i$ and $\mathbf{x}_j$ is defined as the sequence of m ($T \leq m \leq 2T-1$) couples of aligned elements of $\mathbf{x}_i$ and $\mathbf{x}_j$:

$$\boldsymbol{\pi} = ((\pi_1(1), \pi_2(1)), (\pi_1(2), \pi_2(2)), \ldots, (\pi_1(m), \pi_2(m)))$$

where the applications π_1 and π_2, defined from $\{1, \ldots, m\}$ to $\{1, \ldots, T\}$, obey to the following boundary and monotonicity conditions:

(i) $1 = \pi_1(1) \leq \pi_1(2) \leq \ldots \leq \pi_1(m) = T, \quad 1 = \pi_2(1) \leq \pi_2(2) \leq \cdots \leq \pi_2(m) = T$

(ii) $\forall l \in \{1, \ldots, m\}$, $\pi_1(l+1) \leq \pi_1(l) + 1$, $\pi_2(l+1) \leq \pi_2(l) + 1$, $(\pi_1(l+1) - \pi_1(l)) + (\pi_2(l+1) - \pi_2(l)) \geq 1$

Intuitively, an alignment $\boldsymbol{\pi}$ between $\mathbf{x}_i$ and $\mathbf{x}_j$ describes a way to associate each element of $\mathbf{x}_i$ to one or more elements of $\mathbf{x}_j$ and vice-versa. Such an alignment can be conveniently represented by a path in the $T \times T$ grid, where the above monotonicity conditions ensure that the path is neither going back nor jumping. We will denote $\mathcal{A}$ as the set of all alignments between two time series. For measures based on time warp alignments, the integration of a weight vector allows one to weigh differently the different time stamps of the series under consideration. This notion is used for the representative of each cluster $\mathbf{c} = (c_1, \ldots, c_T)$, which is no longer a simple data point (*i.e.*, a simple time series) but rather a time series with an associated weight vector $\mathbf{w} = (w_1, \ldots, w_T)$. The role of the weight vector is to indicate, for each cluster representative, the importance of each time stamp, this importance varying from cluster to cluster. Let d be a dissimilarity measure defined on $\mathbb{R}^T \times (\mathbb{R}^T)^2$, such that d provides a measure of the dissimilarity between a time series $\mathbf{x}$ and a weighted centroid $(\mathbf{c}, \mathbf{w})$. The generalized k-means algorithm aims to find a partition of the data points in K clusters $(C_1, \ldots, C_K)$ such that the intra-cluster dissimilarity is minimized. The associated minimization problem can be written, for $\mathbf{x} \in \mathbf{X}$, as:

$$\arg\min_{\{C_1, \ldots, C_K\}} \sum_{i=1}^{K} \sum_{\mathbf{x} \in C_i} d(\mathbf{x}, (\mathbf{c}_i, \mathbf{w}_i)) \tag{1}$$

where $(\mathbf{c}_i, \mathbf{w}_i) \in \mathbb{R}^T \times \mathbb{R}^T$ is the weighted representative of cluster C_i, defined by:

$$\begin{cases} (\mathbf{c}_i, \mathbf{w}_i) = \arg\min_{\mathbf{c}, \mathbf{w}} \sum_{\mathbf{x} \in C_i} d(\mathbf{x}, (\mathbf{c}, \mathbf{w})) \\ \text{subject to : } \sum_{t=1}^{T} w_t = 1, \quad w_t > 0, \ \forall t \end{cases} \tag{2}$$

where the constraints guarantee that the problem is not degenerate. The generalized k-means algorithm, given in Algorithm 1, is a direct generalization of the standard k-means algorithm in which the centroid update step has been replaced with a more general representative update step that does not rely on the Euclidean distance. The function h (line 6 of Algorithm 1), often referred to as the *representative* function, provides, from a set of time series points, a point in $\mathbb{R}^T$ with an associated weight vector ($h : \mathcal{P}(X) \rightarrow \mathbb{R}^T \times \mathbb{R}^T$). As soon as h satisfies the condition expressed in Inequality 3, the intra-cluster dissimilarity decreases in the representative

Algorithm 1 Generalized k-means.

1: Input: X, K
2: Output: $\{C_1, \ldots, C_K\}$
3: Initialization: $p = 0$; select K representatives $\{\mathbf{c}_1^{(p)}, \ldots, \mathbf{c}_K^{(p)}\}$, either randomly or through strategies *à la* k-means++ [1]
4: **repeat**
5: *Cluster assignment*
$C_i^{(p)} \leftarrow \{\mathbf{x} \in X \mid \mathbf{c}_i^{(p)} = \arg\min_{\mathbf{c}_j, 1 \leq j \leq K} d(\mathbf{x}, (\mathbf{c}_j^{(p)}, \mathbf{w}_j^{(p)}))\}, \quad 1 \leq i \leq K$
6: *Representative update*
$(\mathbf{c}_i^{(p+1)}, \mathbf{w}_i^{(p+1)}) \leftarrow h(C_i^{(p)})$
where the function $h : \mathcal{P}(X) \rightarrow \mathbb{R}^T \times \mathbb{R}^T$ satisfies:
$$\sum_{\mathbf{x} \in C_i^{(p)}} d(\mathbf{x}, h(C_i^{(p)})) \leq \sum_{\mathbf{x} \in C_i^{(p)}} d(\mathbf{x}, (\mathbf{c}_i^{(p)}, \mathbf{w}_i^{(p)})) \tag{3}$$
7: $p \leftarrow p + 1$
8: **until** all clusters $\{C_1, \ldots, C_K\}$ are stable

[1] All our results can however be directly extended to multivariate time series, possibly of different lengths, as the temporal alignments, at the core of the (dis)similarity measures we consider, can be defined in those cases as well.

update step. This decrease is maximum[2] when h is defined using Eq. (2). The intra-cluster dissimilarity also decreases in the assignment step, which is identical to the one used in the standard k-means algorithm. The generalized k-means algorithm provided above thus converges [16–18]. The above formulation can of course be adapted to similarity measures, by substituting the dissimilarity d with a similarity s and *argmin* with *argmax* in Eqs. (1) and (2), and by considering representative estimation functions h satisfying: $\sum_{\mathbf{x}\in C_i^{(p)}} s(\mathbf{x}, h(C_i^{(p)})) \geq \sum_{\mathbf{x}\in C_i^{(p)}} s(\mathbf{x}, (\mathbf{c}_i^{(p)}, \mathbf{w}_i^{(p)}))$. This condition indicates that the solution obtained at iteration p, $h(C_i^{(p)})$, should be better than the one obtained at the previous iteration, $(\mathbf{c}_i^{(p)}, \mathbf{w}_i^{(p)})$.

2.1. Extended time warp measures

We focus in this study on three time warp measures that are commonly used in practice: the dynamic time warping (DTW) [5], which is a dissimilarity measure, the Dynamic Temporal Alignment Kernel (κ_{DTAK}) [6] and the Global Alignment kernel (κ_{GA}) [3,4], which are two similarity measures and constitute a reference in kernel machines in several domains, such as computer vision, speech recognition or machine learning [19–21]. To take into account both global and local temporal differences, we propose an extension of the three measures with a weighted centroid, as discussed above. The extensions mainly introduce a weighted warping function that guides the learned alignments according to the centroid elements importance to capture shared global and local temporal features. The extended form of the dynamic time warping measure is defined as:

$$\text{WDTW}(\mathbf{x}, (\mathbf{c}, \mathbf{w})) = \min_{\boldsymbol{\pi}\in\mathcal{A}} \left(\frac{1}{|\boldsymbol{\pi}|} \sum_{(t',t)\in\pi} w_t^{-\alpha} \, \|x_{t'} - c_t\|^2 \right) \tag{4}$$

in which $w_t^{-\alpha}$ is the introduced non increasing warping function, where $\alpha \in \mathbb{R}^+$ controls the influence of the weighting scheme. The negative exponent guarantees that the most important instants (*i.e.*, those highly weighted) are privileged in the optimal alignment that minimizes the WDTW. For $\alpha = 0$, Eq. (4) leads to the standard unweighted DTW. The complexity of computing WDTW is $O(T^2)$ because the optimal alignment is obtained using dynamic programming. It is, however, possible to be speed it up by considering, instead of $\mathcal{A}$, a subset of alignments usually around the diagonal [22].

More recently, several DTW-based temporal kernels (similarities) that allow one to process time series with kernel machines, *e.g.*, kernel versions of the k-means algorithm, have been introduced,. The most common one is the Dynamic Temporal Alignment Kernel (κ_{DTAK}) [6]. We propose an extension of the κ_{DTAK} pseudo-positive definite kernel, defined as:

$$\text{WK}_{\text{DTAK}}(\mathbf{x}, (\mathbf{c}, \mathbf{w})) = \max_{\boldsymbol{\pi}\in\mathcal{A}} \left(\frac{1}{|\pi|} \sum_{(t',t)\in\boldsymbol{\pi}} w_t^{\alpha} \, e^{-\frac{1}{2\sigma^2}\|x_{t'} - c_t\|^2} \right) \tag{5}$$

where a Gaussian kernel ($e^{-\frac{1}{2\sigma^2}\|x-y\|^2}$) with an associated free parameter σ corresponding to the standard deviation is used to measure the similarity between aligned elements. That time, w_t^{α} with α in [0; 1] is a non-decreasing warping function (see Section 3) that ensures that the most important instants are privileged in the optimal alignment. As before, the standard κ_{DTAK} formulation is obtained with $\alpha = 0$. The complexity of WK_{DTAK}, similar to WDTW, is quadratic in T, the optimal alignment being again obtained using dynamic programming. Note also that Eq. (5) can be viewed as a pre-image problem [23] that aims at estimating the pre-image, in the input space, of the barycenter of the data points in the feature space induced by the WK_{DTAK} kernel. Similarly, the extension of the temporal kernel Global Alignment Kernel [4], which defines a true positive definite kernel, under fair conditions, on the basis of all of the alignments $\boldsymbol{\pi} \in \mathcal{A}$, is defined as:

$$\text{WK}_{\text{GA}}(\mathbf{x}, (\mathbf{c}, \mathbf{w})) = \sum_{\boldsymbol{\pi}\in\mathcal{A}} \prod_{(t',t)\in\boldsymbol{\pi}} w_t^{\alpha} \, e^{-\lambda\Phi(x_{t'}, c_t)} \tag{6}$$

$$\Phi(x_{t'}, c_t) = \left(\frac{1}{2\sigma^2}\|x_{t'} - c_t\|^2 + \log(2 - e^{-\frac{1}{2\sigma^2}\|x_{t'} - c_t\|^2}) \right)$$

where, similar to WK_{DTAK}, a Gaussian kernel is used as the similarity measure between aligned elements. α plays the same role as before and lies again in [0, 1]. The parameter λ is used to attenuate the problem of diagonal dominance that mainly arises for time series of significantly different lengths. Although diagonal dominance leads to positive definite Gram matrices, it degrades the performance of the measure in practice. Finally, the complexity of WK_{GA} is similar to that of WDTW and WK_{DTAK} as one can rely on a recurrence formula to estimate the cost of a given pair (t', t) on the basis of the costs of the three previous pairs $(t'-1, t)$, $(t'-1, t-1)$ and $(t', t-1)$ [3,4]. As for WDTW the time computation can be reduced for WK_{DTAK} by considering only a subset of alignments; for WK_{GA}, a fastest version exists based on a pairing of Gaussian and triangular kernels [3]. We do not consider these variants here because they usually lead to nearly equivalent or worse results in practice. In the remainder of this study we thus focus on the following three extended (dis)similarity measures for time series: 1) WDTW as defined by Eq. (4), 2) WK_{DTAK} as defined by Eq. (5) and 3) WK_{GA} as defined by Eq. (6).

3. Centroid estimation for time warp measures

We first describe here the general strategy followed to estimate the representatives (or centroids) of a cluster of data points (X), *prior* to studying the solution this strategy leads to for the three extended measures introduced above.

3.1. Representative update through alternative optimization

The problem given in Eq. (2) (and its maximization counterpart for similarity measures) can be solved by alternatively minimizing (or maximizing) for $\mathbf{c}$ and $\mathbf{w}$, with the other one being fixed, and by recomputing the optimal alignment when necessary. This strategy corresponds to the following steps[3]:

1. Set initial values for $\mathbf{w}$ and $\mathbf{c}$ and compute for these values $\Pi^* = \{\boldsymbol{\pi}_{\mathbf{x}}^* / \mathbf{x} \in X\}$, where Π^* denotes the optimal alignments between $\mathbf{c}$ and the time series in X.
2. Compute: $\arg\min_{\mathbf{c}} \Sigma_{\mathbf{x}\in X} d(\mathbf{x}, (\mathbf{c}, \mathbf{w}))$ (resp. $\arg\max_{\mathbf{c}} \Sigma_{\mathbf{x}\in X} s(\mathbf{x}, (\mathbf{c}, \mathbf{w}))$).
3. Compute: $\arg\min_{\mathbf{w}} \Sigma_{\mathbf{x}\in X} d(\mathbf{x}, (\mathbf{c}, \mathbf{w}))$ (resp. $\arg\max_{\mathbf{w}} \Sigma_{\mathbf{x}\in X} s(\mathbf{x}, (\mathbf{c}, \mathbf{w}))$), subject to: $\sum_{t=1}^{T} w_t = 1$ and $w_t > 0, \ \forall t$.
4. Compute Π^* for the values of $\mathbf{c}$ and $\mathbf{w}$ obtained previously.
5. Go back to step 2 till $\mathbf{c}$ and $\mathbf{w}$ are stable.

Note that for WK_{GA}, Π^* is not needed because this kernel is based on the sum of all alignments. Steps 2–4 lead to a decrease in the overall dissimilarity between the weighted centroid ($\mathbf{c}$, $\mathbf{w}$) considered at each iteration and the data points in X. Indeed, each

[2] Depending on the dissimilarity measure used, it may not be possible to obtain the point that minimizes Eq. (2), and "looser" functions, based on Inequality 3, have to be considered.

[3] We give here the minimization version, the maximization one being straightforwardly derived from it.

step respectively aims at finding values of $\mathbf{c}$, $\mathbf{w}$ and $\boldsymbol{\pi}$ that minimize the objective function, the other quantities being fixed. The above strategy thus defines an algorithm that converges, and the associated function, which from X produces $(\mathbf{c}, \mathbf{w})$, is a valid representative function. The minimum in steps 2 and 3 can be obtained by computing the partial derivatives w.r.t $\mathbf{c}$ and $\mathbf{w}$ and either setting those partial derivatives to 0 and solving for $\mathbf{c}$ and $\mathbf{w}$ or using gradient descent approaches. Depending on the convexity properties of the measures, the minimum in steps 2 and 3 may be local. This is not a problem *per se* because the procedure still converges while decreasing the inertia between the current pair $(\mathbf{c}, \mathbf{w})$ and the data points. We now study the application of this strategy to each of the time warp measures retained.

3.2. Solution for WDTW

Given $\mathbf{w} = (w_1, \ldots, w_T)$ and $\Pi^* = \{\boldsymbol{\pi}^*_{\mathbf{x}} / \mathbf{x} \in X\}$, the function defined in Eq. (4) is convex in $\mathbf{c}$ and the centroid $\mathbf{c}$ that minimizes the sum of cluster dissimilarities is obtained by solving the partial derivative equation, leading to[4]:

$$\forall t,\ 1 \le t \le T,\ c_t = \frac{\sum_{\mathbf{x} \in X} \frac{1}{|\boldsymbol{\pi}^*_{\mathbf{x}}|} \sum_{(t',t) \in \boldsymbol{\pi}^*_{\mathbf{x}}} x_{t'}}{\sum_{\mathbf{x} \in X} \frac{|N(t,\mathbf{x})|}{|\boldsymbol{\pi}^*_{\mathbf{x}}|}} \tag{7}$$

where $\boldsymbol{\pi}^*_x$ denotes the optimal alignment for $\mathbf{x} \in X$ and $|N(t,\mathbf{x})| = \{t' / (t',t) \in \boldsymbol{\pi}^*_{\mathbf{x}}\}$ denotes the number of time instants of $\mathbf{x}$ aligned to time t of $\mathbf{c}$. The solution for the weights $\mathbf{w}$ is obtained in the same way (the function being, this time, convex in $\mathbf{w}$) by equating the partial derivative of the Lagrangian, integrating the constraints on $\mathbf{w}$ to 0 and solving for $\mathbf{w}$, leading to[4] $\forall t, 1 \le t \le T$:

$$w_t = \frac{A_t^{\frac{1}{1+\alpha}}}{\sum_{t=1}^{T} A_t^{\frac{1}{1+\alpha}}},\ \text{with } A_t = \sum_{\mathbf{x} \in X} \frac{1}{|\boldsymbol{\pi}^*_{\mathbf{x}}|} \sum_{(t',t) \in \boldsymbol{\pi}^*_{\mathbf{x}}} (x_{t'} - c_t)^2 \tag{8}$$

The solution to Eq. (2), corresponding to the representative update step of the generalized k-means algorithm (Algorithm 1), is thus finally obtained through Algorithm 2.

Algorithm 2 *Centroid*$_{\text{WDTW}}$.

1: Initialization: $\mathbf{c}^{(0)}$ randomly selected from X; $\mathbf{w}^{(0)} = (\frac{1}{T}, \ldots, \frac{1}{T})$; $\Pi^{*(0)}$: optimal alignments between X and $(\mathbf{c}^{(0)}, \mathbf{w}^{(0)})$; $p = 0$
2: **repeat**
3: $p \leftarrow p + 1$
4: Update $c_t^{(p)}$ using Eq. (7), $1 \le t \le T$
5: Update $w_t^{(p)}$ using Eq. (8), $1 \le t \le T$
6: Update $\Pi^{*(p)}$: optimal alignments between X and $(\mathbf{c}^{(p)}, \mathbf{w}^{(p)})$
7: **until** $(\mathbf{c}^{(p)}, \mathbf{w}^{(p)}) \approx (\mathbf{c}^{(p-1)}, \mathbf{w}^{(p-1)})$

3.3. Solution for WK$_{\text{DTAK}}$

Given $\mathbf{w} = (w_1, \ldots, w_T)$ and $\Pi^* = \{\boldsymbol{\pi}^*_{\mathbf{x}} / \mathbf{x} \in X\}$, the function defined in Eq. (5) is convex in $\mathbf{c}$. The partial derivative equation for $\mathbf{c}$ for κ_{DTAK} is, $\forall t, 1 \le t \le T$:

$$\frac{\partial L}{\partial c_t} = \sum_{\mathbf{x} \in X} \frac{1}{|\boldsymbol{\pi}^*_{\mathbf{x}}|} \sum_{(t',t) \in \boldsymbol{\pi}^*_{\mathbf{x}}} w_t^{\alpha} \frac{(x_{t'} - c_t)}{\sigma^2} e^{\left(-\frac{(x_{t'} - c_t)^2}{2\sigma^2}\right)} = 0 \tag{9}$$

where L is given by Eq. (5). We know of no closed-form solution to the above equation and resort, in this study, to a gradient ascent method based on the following update rules at iteration p:

$$c_t^{(p+1)} = c_t^{(p)} + \eta^{(p)} \frac{\partial L}{\partial c_t^{(p)}} \text{ and } \eta^{(p+1)} = \frac{\eta^{(p)}}{p}\ (\eta^{(0)} = 1)$$

[4] We omit the derivation which is purely technical.

Algorithm 3 *Centroid*$_{\text{WK}_{\text{DTAK}}}$

1: Initialization: $\mathbf{c}^{(0)}$ randomly selected from X; $\mathbf{w}^{(0)} = (\frac{1}{T}, \ldots, \frac{1}{T})$; $\Pi^{*(0)}$: optimal alignments between X and $(\mathbf{c}^{(0)}, \mathbf{w}^{(0)})$; $p = 0$
2: **repeat**
3: $p \leftarrow p + 1$
4: // Update $c_t^{(p)}$, $1 \le t \le T$:
5: $q = 0,\ \eta = 1$
6: **repeat**
7: $q \leftarrow (q + 1),\ \eta \leftarrow \frac{\eta}{q}$
8: $c^{(q+1)} \leftarrow c^{(q)} + \eta \frac{\partial L}{\partial c}$, with $\frac{\partial L}{\partial c}$ given by Eq. (9)
9: **until** $\frac{\partial L}{\partial c} \approx 0$
10: Update $w_t^{(p)}$ using Eq. (10), $1 \le t \le T$
11: Update $\Pi^{*(p)}$: optimal alignments between X and $(\mathbf{c}^{(p)}, \mathbf{w}^{(p)})$
12: **until** $(\mathbf{c}^{(p)}, \mathbf{w}^{(p)}) \approx (\mathbf{c}^{(p-1)}, \mathbf{w}^{(p-1)})$

For the weights estimation, the function defined in Eq. (5) is convex in $\mathbf{w}$ as $\alpha \in [0; 1]$. The solution of the partial derivative equation, obtained by equating the partial derivative of the Lagrangian, integrating the constraints on $\mathbf{w}$ to 0, can easily be obtained. Given $\mathbf{c} = (c_1, \ldots, c_T)$ and $\Pi^* = \{\boldsymbol{\pi}^*_{\mathbf{x}} / \mathbf{x} \in X\}$, the weight vector $\mathbf{w}$ that maximizes the sum of intra-cluster similarities subject to $\sum_{t=1}^{T} w_t = 1$ and $w_t > 0$, $\forall t$, is defined by:

$$w_t = \frac{A_t^{\frac{1}{1-\alpha}}}{\sum_{t=1}^{T} A_t^{\frac{1}{1-\alpha}}} \text{ with } A_t = \sum_{\mathbf{x} \in X} \frac{1}{|\boldsymbol{\pi}^*_{\mathbf{x}}|} \sum_{(t',t) \in \boldsymbol{\pi}^*_{\mathbf{x}}} e^{\left(-\frac{(x_{t'} - c_t)^2}{2\sigma^2}\right)} \tag{10}$$

Algorithm 3 summarizes the steps required to solve the centroid estimation problem (Eq. (2)) for WK$_{\text{DTAK}}$. The gradient ascent steps for estimating $\mathbf{c}$ correspond to lines 2–9.

3.4. The case of WK$_{\text{GA}}$

Given $\mathbf{w} = (w_1, \ldots, w_T)$, the partial derivative of WK$_{\text{GA}}$ for updating the centroid $\mathbf{c}$ takes the form:

$$\frac{\partial L}{\partial c_t} = \sum_{\mathbf{x} \in X} \sum_{\boldsymbol{\pi} \in \mathcal{A}} \left(\sum_{(t',t) \in \boldsymbol{\pi}} - \frac{(x_{t'} - c_t)}{\sigma^2\ (2 - e^{\frac{-1}{2\sigma^2}(x_{t'} - c_t)^2})} \right) \times \tag{11}$$

$$\prod_{(t',t) \in \boldsymbol{\pi}} w_t^{\alpha}\ e^{-\lambda \Phi(x_{t'}, c_t)}$$

where $\Phi(x_{t'}, c_t)$ and L correspond to Eq. (6). We know of no closed-form solution to the equation $\frac{\partial L}{\partial c_t} = 0$. More importantly, the computation of the derivative given in Eq. (11) cannot benefit from the recurrence formulas of WK$_{\text{GA}}$, which ensure an efficient computation of the measure. Thus, if one wants to use gradient ascend methods, one needs to compute the scores associated with each alignment and sum them, which is impractical in situations where T is large (more than a few tens). For this reason, we will use WK$_{\text{GA}}$ in kernel k-means only.

4. Experiments

In this section, we first describe the datasets retained to conduct our experiments *prior* to comparing the generalized k-means algorithms, based on the extended WDTW(Eq. (4)) and WK$_{\text{DTAK}}$ (Eq. (5)) and the centroid estimations given in Section 3, to two alternative approaches i) k-medoids with the standard unweighted DTW and ii) kernel k-means with the standard unweighted κ_{DTAK} and κ_{GA} temporal kernels.

Table 1
Data description.

	Class Nb. *K*	Size *n*	TS. length *T*	Isotropy (*p*-Value, std.)	Isolation *Ratio*
BEEF	5	30	470	(0.37, 0.39)	0.06
BME	3	150	90	**(0.00, 0.00)**	**0.27**
CBF	3	930	128	**(0.00, 0.00)**	**0.22**
CC	6	600	60	**(0.00, 0.01)**	0.02
COFFEE	2	28	286	(0.44, 0.26)	**0.50**
CONSSEASON	2	365	144	(0.44, 0.43)	**0.45**
ECG200	2	100	96	**(0.09, 0.02)**	**0.49**
FACEFOUR	4	88	350	(0.26, 0.43)	0.18
FISH	7	175	463	(0.53, 0.26)	0.13
GUNPOINT	2	150	150	**(0.14, 0.20)**	**0.48**
LIGHTING2	2	61	637	0.28, 0.39)	**0.48**
LIGHTING7	7	73	319	(0.29, 0.30)	0.08
MEDICALIMAGES	10	760	99	**(0.09, 0.17)**	**0.28**
OLIVEOIL	4	30	570	(0.39, 0.35)	0.15
OSULEAF	6	242	427	(0.49, 0.34)	0.16
SWEDISHLEAF	15	625	128	**(0.15, 0.30)**	0.03
SYMBOLES	6	180	398	**(0.06, 0.11)**	0.00
TRACE	4	200	275	**(0.15, 0.23)**	0.00
TWOPATTERNS	4	400	128	(0.23, 0.45)	0.09
UMD	3	150	121	**(0.00, 0.00)**	**0.20**

4.1. Datasets

Our experiments are conducted on 20 public datasets.[5] The classes composing the datasets are known beforehand and define the ground truth partition. To characterize these datasets, and because clustering methods such as *k*-means are known to be more efficient when the clusters are isotropic (*i.e.*, spherical) and well isolated, we computed, for each dataset, the Bartlett's test of sphericity[6] [24,25] as well as the isolation ratio. The isotropy is measured as the average of Bartlett's *p*-value on the clusters, as the *p*-value corresponds to the probability of being spherical. The cluster isolation ratio corresponds to the ratio of the sum of the WDTW dissimilarities within clusters to the total WDTW dissimilarities. The lower the isolation ratio, the more isolated the clusters are. Table 1 describes the datasets considered, with their main characteristics: number of clusters (K), dataset size (n), time series length (T), *p*-Values for the isotropy and an isolation ratio. In particular, *p*-Values ≤ 0.2 indicative of non-isotropic datasets, and *isolation ratio* ≥ 0.2, indicative of non-well-isolated datasets, are in bold. As one can note from the *p*-Values and isolation ratios displayed in Table 1, most of the datasets considered are composed of clusters that are either non-isotropic or not well-isolated (or both) and are thus challenging for the *k*-means and kernel *k*-means algorithms [13]. Furthermore, to visualize the underlying structure of the datasets, we performed a multidimensional scaling[7] [26] on the WDTW pairwise dissimilarity matrix. Fig. 1 displays the cluster structures on the first plan; with a stress lower than 20%, the representations obtained can be considered as accurate images of the underlying structures (the stress being the error between the distances induced by the first plan and the original dissimilarities). As one can note, the datasets retained have very diverse structures and shapes and many of them are not linearly separable.

4.2. Clustering methods

We compare here the generalized *k*-means algorithms proposed (denoted as G*k*-means) with *k*-medoids (*k*-med) under the standard DTW and kernel *k*-means[8] (K*k*-means) based on the standard κ_{DTAK} and κ_{GA}. We focus on these methods because they constitute the most frequently used variants of the *k*-means approach for time series clustering. As mentioned before, these variants are mainly used to bypass the centroid estimation process required by *k*-means with time warp measures. For our comparison, we rely on the Rand index[9] [27], which measures the agreement between the obtained partition and the ground truth one, to assess each method. The higher the Rand index (which lies in [0, 1]), the better the agreement is. In particular, the maximal value, 1, is reached when the partitions are identical. For all methods, the parameters are estimated using a validation set through a standard line/grid search process, as described in Table 2 (50% of the data are used to compute the different models and 50% are used for model selection; the parameters retained are the ones that maximize the Rand index on the validation set). Finally, the results reported hereafter are averaged through a bootstrap process, with 10 repetitions. Because κ_{DTAK} is not a strictly definite positive kernel [3,4], we systematically added a scaled identity matrix δI to the Gram matrices with negative eigenvalues, where δ is the absolute value of the smallest negative eigenvalue and I is the identity matrix. For κ_{GA},

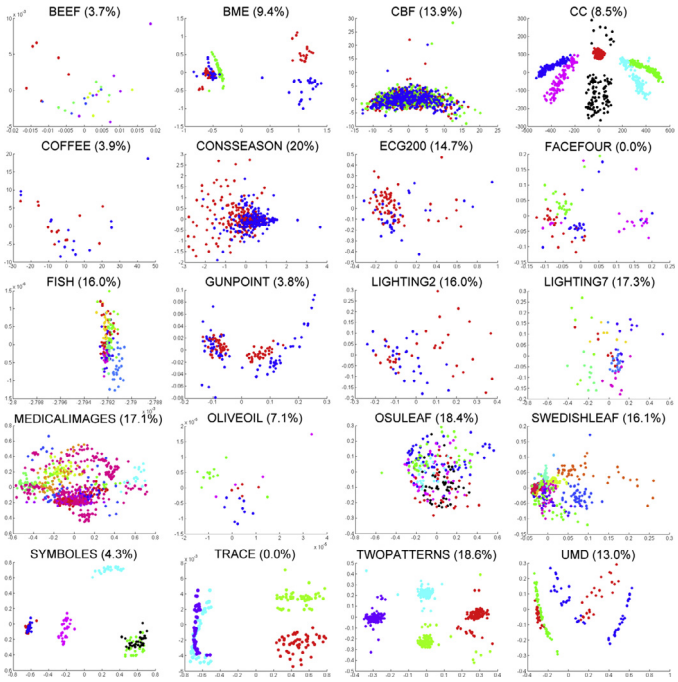

Fig. 1. Structures underlying datasets.

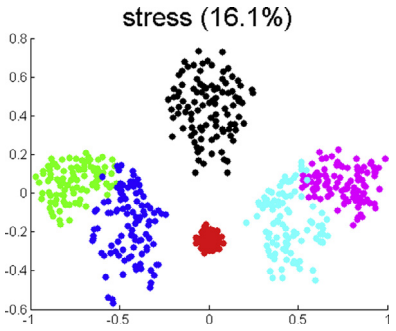

Fig. 2. CC clusters based on WK_{DTAK} dissimilarities.

[5] CONSSEASON available at http://archive.ics.uci.edu/ml/datasets/Individual+household+electric+power+consumption, UMD, BME at http://ama.liglab.fr/~douzal/tools.html, the rest of the data at http://www.cs.ucr.edu/~eamonn/time_series_data/

[6] *Barspher* Matlab function.

[7] *mdscale* Matlab function.

[8] *kmedoids*, *kernelkmeans* Matlab functions.

[9] *RandIndex* Matlab function.

Table 2
Parameter line/grid: **med**(x) stands for the empirical median of x evaluated on the validation set. The . multiplication is element-wise, *e.g.*, $\{1, 2, 3\}$. **med** $= \{$**med**, 2**med**, 3**med**$\}$. In $\|x - y\|$, x and y are vectors sampled randomly within time series in the validation set.

Method	Metric	Line/grid values
Kernel *k*-means	K_{DTAK}	$\sigma \in \{0.2, 0.5, 1, 2, 5\}$. **med**$(\|x - y\|)$
	K_{GA}	$\sigma \in \{0.2, 0.5, 1, 2, 5\}$. **med**$(\|x - y\|)$ $\lambda \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10\}$
Generalized *k*-means	WDTW	$\alpha \in \{10^{-2}, 10^{-1}, 10^{0}\}$
	WK_{DTAK}	$\alpha \in \{10^{-2}, 10^{-1}, 10^{0}\}$ $\sigma \in \{0.2, 0.5, 1, 2, 5\}$. **med**$(\|x - y\|)$

Table 3
Rand index.

	Kernel *k*-means (K*k*-means)		*K*-medoid (*K*-med)	Generalized *k*-means (G*k*-means)	
	k_{DTAK}	k_{GA}	DTW	WDTW	WK_{DTAK}
BEEF	0.735	0.737	0.710	**0.752**	0.738
BME	0.390	0.377	0.224	**0.446**	0.403
CBF	0.710	0.741	0.701	**0.776**	0.756
CC	0.591	0.583	0.804	**0.832**	0.808
COFFEE	0.505	0.547	0.484	**0.605**	0.563
CONSSEASON	0.566	0.522	0.528	**0.744**	0.713
ECG200	0.578	0.590	0.519	**0.658**	0.608
FACEFOUR	0.744	0.697	0.714	**0.816**	0.774
FISH	0.828	0.844	0.768	**0.893**	0.839
GUNPOINT	0.502	0.497	0.497	0.719	**0.761**
LIGHTING2	0.503	0.496	0.563	**0.718**	0.540
LIGHTING7	0.827	0.815	0.816	**0.881**	0.835
MEDICALIMAGES	0.688	0.683	0.682	**0.866**	0.699
OLIVEOIL	0.832	0.826	0.793	**0.848**	0.833
OSULEAF	0.752	0.772	0.730	**0.914**	0.726
SWEDISHLEAF	0.892	0.841	0.891	**0.909**	0.896
SYMBOLES	0.672	0.679	0.783	**0.912**	0.788
TRACE	0.666	0.625	0.652	**0.707**	0.676
TWOPATTERNS	0.696	0.684	0.842	**0.947**	0.852
UMD	0.307	0.292	0.261	**0.485**	0.422

the estimated value of the regularization parameter $\lambda \approx 1$ shows that the problem of diagonal dominance has not been encountered for the datasets retained.

4.3. Analysis of the results

The Rand indices and time consumptions for each method, on each dataset are, respectively, reported in Tables 3 and 4. Results in bold correspond to results for which the difference is statistically significantly better (t-test at 1% risk) compared with the non-bold results. Based on the Rand indices displayed in Table 3, one can note that G*k*-means with WDTW leads to the best clustering results overall (19 datasets out of 20), followed by G*k*-means with WK_{DTAK}. Furthermore, the clustering methods based on G*k*-means introduced here outperform *k*-medoids and kernel *k*-means on all datasets.

4.3.1. Impact of isotropy and isolation on clustering quality

The most challenging datasets are undeniably UMD and BME, with the lowest Rand indices varying in [0.22, 0.39] for *k*-medoids and kernel *k*-means and in [0.40, 0.44] for G*k*-means methods. UMD and BME, in addition to being composed of weakly isolated and weakly isotropic classes (Table 1), include time series of distinct global behaviors within clusters, as discussed in [15]. They thus necessitate a dissimilarity or similarity measure able to capture the commonly shared features within clusters. G*k*-means, by relying on weighted centroids, can capture the characteristics shared locally within clusters and improves the clustering performances over the standard approaches, which fail to capture such local features. The second category of challenging datasets consists of COFFEE, CONSSEASON, ECG200, GUNPOINT, and LIGHTING2 and is composed of weakly isolated but highly or mildly isotropic clusters (Table 1). Both *k*-medoids and kernel *k*-means lead to weak clustering results with a Rand index lying in [0.48, 0.59], whereas G*k*-means (with WDTW) leads to notably better clustering results with a Rand index in [0.60, 0.74]. A third category of datasets is defined by CBF and MEDICALIMAGES and is composed of mildly isolated but weakly isotropic clusters. If both *k*-medoids and kernel *k*-means yield reasonable clustering results, with a Rand index in [0.68, 0.70], G*k*-means (with WDTW) yields the best clustering results with a Rand index in [0.77, 0.86]. For the remaining datasets, composed of highly isolated and mildly isotropic clusters, good clustering results (Rand index in [0.59, 0.89]) are obtained using *k*-medoids and kernel *k*-means, but are improved significantly by using G*k*-means (with WDTW) with a Rand index in [0.70, 0.94]. As one can note from the above results, both kernel *k*-means and *k*-medoids provide relatively good clustering results even when the clusters are non-spherical; they are, however, limited when the clusters are non-isolated. On the other hand, G*k*-means with WDTW obtains very good clustering results on all datasets, particularly on challenging datasets composed of clusters that are both non-isotropic and not well-isolated.

Table 4
Time consumption(s).

	Kernel *k*-means (K*k*-means)		*K*-medoid (*K*-med)	Generalized *k*-means (G*k*-means)	
	k_{DTAK}	k_{GA}	DTW	WDTW	WK_{DTAK}
BEEF	53.8	77.4	**49.8**	58.7	76.3
BME	37.4	42.1	32.6	**28.5**	36.7
CBF	3003.9	3090.8	2442.9	**395.8**	633.4
CC	237.0	295.6	221.7	**85.5**	261.8
COFFEE	18.8	17.1	16.8	**15.7**	23.3
CONSSEASON	594.9	647.5	504.3	**125.0**	484.0
ECG200	12.6	26.4	15.3	**10.2**	12.2
FACEFOUR	363.8	343.4	**206.7**	292.6	320.5
FISH	1103.5	2520.8	1525.5	**663.1**	1007.1
GUNPOINT	52.1	150.5	85.63	**39.4**	45.5
LIGHTING2	303.8	613.8	397.1	**141.9**	158.5
LIGHTING7	**94.6**	191.4	112.3	98.4	109.5
MEDICALIMAGES	943.6	1581.4	940.2	**236.2**	1228.7
OLIVEOIL	54.4	116.9	67.2	**52.2**	73.8
OSULEAF	1784.7	3979.6	1845.8	**787.6**	971.8
SWEDISHLEAF	753.6	1741.3	1083.5	**350.8**	659.9
SYMBOLES	938.2	1233.3	912.4	**573.9**	777.5
TRACE	595.7	762.0	573.8	**283.9**	456.6
TWOPATTERNS	320.9	617.3	452.5	**110.2**	172.9
UMD	50.0	73.9	53.9	**37.7**	44.9

4.3.2. Complexity and time considerations

Table 4 displays the time consumptions of the different methods. As one can note, G*k*-means is the fastest clustering approach, with a nearly constant time requirement for all of the methods on small datasets (*e.g.*, BEEF, BME, COFFEE, ECG200). For large datasets (*e.g.*, CBF, CC, MEDICALIMAGES, SWEDISHLEAF), both K*k*-means and *k*-med are significantly slower mainly due to the involved pairwise comparisons, whereas G*k*-means, particularly with WDTW, remains remarkably fast. These empirical results are in agreement with the theoretical complexities of each method. Indeed, let n be the number of time series to be clustered in K clusters and let p denote the complexity of the measure between aligned instants ($p = 5$ for WDTW $p = 8$ for WK_{DTAK} and $p = 18$ for WK_{GA} – see Eqs. (4)–(6)). Then, the complexity of G*k*-means is $O(pnKT^2)$, the one of *k*-medoids is $O(pn(n-K)T^2)$ and the one of kernel *k*-means is

$O(pn^2T^2)$, showing that kernel k-means is slower than k-medoids, which is in turn slower than Gk-means; the difference between Gk-means with wDTW and κ_{DTAK} is mainly due to the ascent gradient part, while the difference between Kk-means with κ_{GA} and κ_{DTAK} is explained by the different values of p for each method.

5. Conclusion

This work introduces a generalized centroid-based clustering algorithm for temporal data under time warp measures. For this, we propose i) an extension of the common time warp measures and ii) a tractable, fast and efficient estimation of the cluster representatives, under the extended time warp measures, that captures local temporal features. The efficiency of this algorithm is analyzed on a wide range of challenging datasets, which are non-isotropic (*i.e.*, non-spherical), not well-isolated and linearly non-separable, and contrasted to the behaviors of k-medoids and kernel k-means. The results show the benefits of the method proposed, which, in addition to being faster, outperforms the other methods on all datasets. Detailed analyses of the impact of isotropy and isolation of the clusters on the effectiveness of the studied clustering methods are finally provided.

References

[1] D. Arthur, S. Vassilvitskii, K-means++: the advantages of careful seeding, in: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '07, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2007, pp. 1027–1035.

[2] C. Bahlmann, B. Haasdonk, H. Burkhardt, Online handwriting recognition with support vector machines-a kernel approach, in: Proceedings of the Eighth International Workshop on Frontiers in Handwriting Recognition, 2002, IEEE, 2002, pp. 49–54.

[3] M. Cuturi, Fast global alignment kernels, in: Proceedings of the 28th International Conference on Machine Learning (ICML-11), 2011, pp. 929–936.

[4] M. Cuturi, J.-P. Vert, Ø. Birkenes, T. Matsui, A kernel for time series based on global alignments, in: Proceedings of the International Conference on Acoustics, Speech and Signal Processing, 11, 2007, pp. 413–416.

[5] J. Kruskall, M. Liberman, The symmetric time warping algorithm: from continuous to discrete, Time Warps, String Edits and Macromolecules, Addison-Wesley, 1983.

[6] H. Shimodaira, K.-i. Noma, M. Nakai, S. Sagayama, Dynamic time-alignment kernel in support vector machine, in: Proceedings of Neural Information Processing Systems, NIPS, 14, 2002, pp. 921–928.

[7] C. Notredame, D.G. Higgins, J. Heringa, T-coffee: a novel method for fast and accurate multiple sequence alignment, J. Mol. Biol. 302 (1) (2000) 205–217.

[8] J.D. Thompson, D.G. Higgins, T.J. Gibson, Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice, Nucleic Acids Res. 22 (22) (1994) 4673–4680.

[9] W.H. Abdulla, D. Chow, G. Sin, Cross-words reference template for dtw-based speech recognition systems, in: Proceedings of Conference on Convergent Technologies for the Asia-Pacific Region, TENCON 2003, 4, IEEE, 2003, pp. 1576–1579.

[10] C. Notredame, D. Higgins, J. Heringa, T-coffee: a novel method for fast and accurate multiple sequence alignment, J. Mol. Biol. 302 (1) (2000) 205–217.

[11] F. Petitjean, A. Ketterlin, P. Gançarski, A global averaging method for dynamic time warping, with applications to clustering, Pattern Recognit. 44 (3) (2011) 678–693.

[12] I.S. Dhillon, Y. Guan, B. Kulis, Kernel k-means: spectral clustering and normalized cuts, in: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2004, pp. 551–556.

[13] M. Girolami, Mercer kernel-based clustering in feature space, IEEE Trans. Neural Netw. 13 (3) (2002) 780–784.

[14] W. Liao, Clustering of time series data – a survey., Pattern Recognit. 38 (2005) 1857–1874.

[15] C. Frambourg, A. Douzal-Chouakria, E. Gaussier, Learning multiple temporal matching for time series classification, in: A. Tucker, F. Höppner, A. Siebes, S. Swift (Eds.), Intelligent Data Analysis, Springer Berlin Heidelberg, London, 2013, pp. 198–209.

[16] S.Z. Selim, M.A. Ismail, K-means-type algorithms: a generalized convergence theorem and characterization of local optimality, IEEE Trans. Pattern Anal. Mach. Intell. 1 (1984) 81–87.

[17] T. Hastie, R. Tibshirani, J. Friedman, J. Franklin, The elements of statistical learning: data mining, inference and prediction, Math. Intell. 27 (2) (2005) 83–85.

[18] H.-H. Bock, Origins and extensions of the k-means algorithm in cluster analysis, Electron. J. Hist. Prob. Stat. 4 (2008) 1–18.

[19] H. Shimodaira, K.-i Noma, Dynamic time-alignment kernel in support vector machine, Adv. Neural Inf. Process. Syst. 14 (2002) 921.

[20] W. Bailer, Sequence Kernels for Clustering and Visualizing Near Duplicate Video Segments, Springer, 2012.

[21] F. Zhou, F. De la Torre, J.K. Hodgins, Hierarchical aligned cluster analysis for temporal clustering of human motion, IEEE Trans. Pattern Anal. Mach. Intell. 35 (3) (2013) 582–596.

[22] H. Sakoe, S. Chiba, Dynamic programming algorithm optimization for spoken word recognition, IEEE Trans. Acoust. Speech Signal Process. 26 (1) (1978) 43–49.

[23] J.T.-Y. Kwok, I.W.-H. Tsang, The pre-image problem in kernel methods, IEEE Trans. Neural Netw. 15 (6) (2004) 1517–1525.

[24] W.W. Cooley, P.R. Lohnes, Multivariate Data Analysis, John Wiley, 1971.

[25] D.A. Jackson, Stopping rules in principal components analysis: a comparison of heuristical and statistical approaches, Ecology 74 (8) (1993) 2204–2214.

[26] T. Cox, M. Cox, Multidimensional Scaling, Chapman and Hall, 2001.

[27] W.M. Rand, Objective criteria for the evaluation of clustering methods, J. Am. Stat. Assoc. 66 (336) (1971) 846–850.