



HAL
open science

Learning temporal alignments for multivariate time series discrimination

Cédric Frambourg, Ahlame Douzal-Chouakria, Éric Gaussier, Jacques Demongeot

► **To cite this version:**

Cédric Frambourg, Ahlame Douzal-Chouakria, Éric Gaussier, Jacques Demongeot. Learning temporal alignments for multivariate time series discrimination. 2012, 14 p. hal-00744747v1

HAL Id: hal-00744747

<https://hal.science/hal-00744747v1>

Submitted on 23 Oct 2012 (v1), last revised 6 May 2013 (v2)

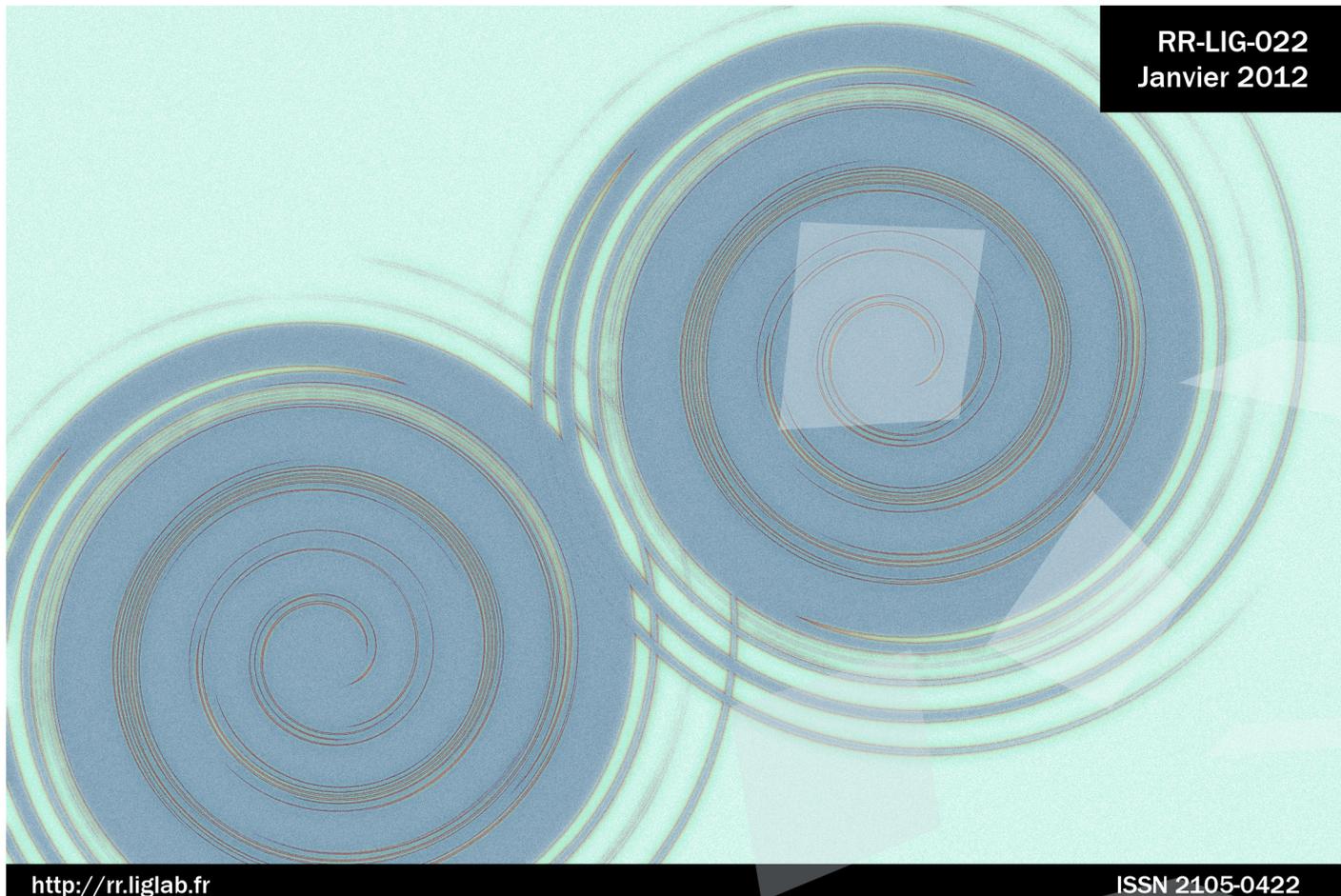
HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Les rapports de recherche du LIG

Learning temporal alignments for multivariate time series discrimination

Cédric FRAMBOURG, PhD Student, LIG, Grenoble University (UJF / IGA), France
Ahrame DOUZAL-CHOUAKRIA, Associate Professor, LIG, Grenoble University (UJF), France
Éric GAUSSIER, Associate Professor, LIG, Grenoble University (UJF), France
Jacques DEMONGEOT, Professor, TIMC, Grenoble University (UJF), France



RR-LIG-022
Janvier 2012

Learning temporal alignments for multivariate time series discrimination

C. Frambourg, A. Douzal-Chouakria, E. Gaussier
UJF-Grenoble 1/CNRS, Université de Grenoble, LIG UMR 5217/AMA team
UJF-Grenoble 1/CNRS, TIMC-IMAG UMR 5525.
Email: {Cedric.Frambourg, Ahlame.Douzal, Eric.Gaussier}@imag.fr
J. Demongeot
UJF-Grenoble 1/CNRS, Université de Grenoble, TIMC UMR 5525
Email: Jacques.Demongeot@imag.fr

Abstract—For time series discrimination, this paper proposes a new approach to align time series with respect to the commonly shared features within classes and the most differential ones between classes. The main idea behind the proposed approach is to use a variance/covariance criterion to strengthen or weaken aligned observations according to their contribution to the variability within and between classes. To this end, the classical variance/covariance expression is extended to a set of time series, as well as to a partition of time series based on classes. A new algorithm is then proposed to learn alignments between time series so as to minimize the within variance and maximize the between variance. The relevance of the learned alignments is studied through a nearest neighbor time series classification on real and synthetic datasets. The carried out experiments reveal that the proposed approach is able to capture fine-grained distinctions between time series across classes, and outperforms standard approaches on several different datasets, all the more so that the correspondence between time series within the same class is complex.
Keywords: time series alignment, discriminant analysis, variance-covariance, time series classification, dynamic time warping, Euclidean distance

I. INTRODUCTION

Time series originating from the same (or similar) sources are often noisy as the timing of salient events can be extremely variable. For example, in the context of electric networks, a particular peak associated with the same underlying event may appear at different times, depending on the use of the plugs monitored. To allow time series comparison while dealing with time delays, numerous alignment strategies have been proposed, as the ones based on Dynamic Time Warping (DTW) (e.g. [1], [2], [3], [4], [5]), which however yields a very local view as the alignment depends only on the couple of time series under consideration; furthermore, the process of alignment is decoupled from the one of analysis (as clustering or classification), weakening the use of the alignment in real applications.

To partly overcome these problems, Gaffney et al. [6] propose a probabilistic framework to jointly handle the clustering and the alignment processes. However, the proposed alignments are limited to time series of a same class so that the discriminative power of the method is limited. In Listgarten et al. [7], a hierarchical Bayesian model is

proposed to perform detection of rare differences between classes of time series. This model allows one to align time series simultaneously across all classes, while detecting and characterizing class-specific differences. Ramsay et al. [8] propose a time series clustering model where an alignment function is learned for each time series, parameterized with order one B-spline coefficients. The learned alignments account for a common shared structure within clusters.

Although these approaches yield more accurate alignments between time series, they all assume that time series within the same class or cluster share a single, common global structure. It is however indisputable that in real applications time series peculiarities may be more complex. For instance, time series of a same class may have distinctive global behaviors while sharing common local features; or one may have some time series of different classes of nearly similar global behaviors as illustrated in Figures 2 and 3, where only small differences appear between classes.

In the context of discriminating complex time series, one thus needs to align time series with respect to the commonly shared features, pertaining to potentially many underlying global structures, within classes, and to identify the most differential features between classes. We propose to do so here through the use of a variance/covariance criterion to strengthen or weaken links according to their contributions to the variances within and between clusters. The variance/covariance measure is a classical criterion, used in many approaches, including discriminant analysis, dimensionality reduction, clustering and classification, and variants of it have already been proposed for graph-structured data (see for example [9], [10], [11], [12]). Its use for learning alignments between time series has however never been investigated before, to our knowledge.

The rest of the paper is organized as follows. In Section II, we propose an extension of the classical variance/covariance expression to a set of time series, then to a partition based on classes of time series. We then present in Section III two new algorithms for learning temporal alignments within and between classes and provide a proof of their convergence. Based on the learned alignments, a discriminative distance is defined in Section IV for time series nearest neighbor

classification. In Section V we evaluate the relevance of the learned alignments to discriminate and classify time series. The obtained results as well as future work are discussed in Section VI.

II. THE VARIANCE/COVARIANCE OF TIME SERIES DATA

We first recall here the definition of the conventional variance/covariance matrix, prior to extend it to a set of time series and then to a partitioned (according to classes) set of time series. Let X be the $(n \times p)$ data matrix providing the description of n observations by p numerical variables. The conventional $(p \times p)$ variance/covariance matrix expression is:

$$V = X^t(I - UP)^t P(I - UP)X \quad (1)$$

where, I is the diagonal identity matrix, U the unit matrix, and P a diagonal matrix of weights, generally set to $p_i = \frac{1}{n}$ for equally weighted observations.

In the case of a set of time series, let X be the $(nT \times p)$ matrix providing the description of n multivariate time series S_1, \dots, S_n by p numerical variables at T time stamps. The general term x_{ij}^l of X gives the value of the variable X_j ($j = 1, \dots, p$) taken by S_l ($l = 1, \dots, n$) at the i th time stamp ($i = 1, \dots, T$). Alignments between n time series can be encoded through a matrix M composed of n^2 block matrices $M^{ll'}$ ($l = 1, \dots, n; l' = 1, \dots, n$). A block $M^{ll'}$ is a $(T \times T)$ matrix that specifies the alignment between S_l and $S_{l'}$, and its general term $m_{ii'}^{ll'} \in [0, 1]$ indicates the intensity of the linkage between the observation of S_l at time i and the observation of $S_{l'}$ at time i' . For example:

- A complete linkage connecting all observations of S_l and $S_{l'}$, whatever their time stamps, is obtained by setting $\forall i, i' \in \{1, \dots, T\}$, $m_{ii'}^{ll'} = \frac{1}{T}$, which corresponds to $M^{ll'} = \frac{1}{T}U_T$, U_T being the $(T \times T)$ unit matrix;
- The euclidean alignment connecting observations that occur at the same time is obtained by setting $\forall i, i' \in \{1, \dots, T\}$, $m_{ii'}^{ll'} = 1$ if $i = i'$ and 0 otherwise, which corresponds to $M^{ll'} = I$;
- A dynamic time warping alignment is obtained by setting $\forall i, i' \in \{1, \dots, T\}$, $m_{ii'}^{ll'} = 1$ if i is aligned with i' by the standard DTW, and 0 otherwise.

Then, the $(p \times p)$ variance/covariance matrix V_M induced by a set of time series S_1, \dots, S_n connected to one another according to the alignment matrix M can be defined on the basis of Eq.1, as:

$$V_M = X^t(I - M)^t P(I - M)X \quad (2)$$

where P is a $(nT \times nT)$ diagonal matrix of weights, with $p_i = \frac{1}{nT}$ for equally weighted observations. The variance of each variable X_j as well as the total variance can thus be defined through the diagonal elements of V_M .

Definition 1: The variance of each variable X_j ($j = 1, \dots, p$) corresponds to the j th diagonal term of V_M and is given by:

$$V_{M_j} = \sum_{l=1}^n \sum_{i=1}^T p_i (x_{ij}^l - \sum_{l'=1}^n \sum_{i'=1}^T m_{ii'}^{ll'} x_{i'j}^{l'})^2 \quad (3)$$

The total variance of the aligned time series is defined by:

$$TotV_M = \sum_{j=1}^p V_{M_j} \quad (4)$$

We now proceed to define the variance within and between classes when the time series are partitioned into classes.

A. The variance induced by a partition of time series

Let us now consider the set of time series S_1, \dots, S_n partitioned into K groups, with $y_i \in \{1, \dots, K\}$ the class label of S_i and n_k the size of class k (i.e. the number of time series contained in class k). In such a case, the within variance/covariance matrix provides a measure of the dispersion of time series within classes, and the within variance a measure of how close time series are within classes. Following the definition for the variance of a set of time series, we define the within variance with an intra-class alignment matrix M as the sum, restricted over the time series of the same class, of the variance of each variable.

Definition 2: The within variance with an intra-class alignment matrix M is given by:

$$TotWV_M = \frac{1}{nT} \sum_{k=1}^K \sum_{j=1}^p \sum_{l=1}^{n_k} \sum_{i=1}^T (x_{ij}^l - \frac{1}{n_k} \sum_{l'=1}^{n_k} \sum_{i'=1}^T m_{ii'}^{ll'} x_{i'j}^{l'})^2$$

with

$$M^{ll'} = \begin{cases} \mathbf{I} & \text{if } l = l' \\ \neq \mathbf{0} & \text{if } y_l = y_{l'} \text{ and } l \neq l' \\ \mathbf{0} & \text{if } y_l \neq y_{l'} \end{cases} \quad (5)$$

where \mathbf{I} and $\mathbf{0}$ are the $(T \times T)$ identity and zero matrices, respectively.

The first setting for M simply states that each time series is aligned with itself at each time stamp, which ensures that the variance of a time series compared to itself is zero. The second setting simply states that two time series *within the same class* are connected, with values which are not necessarily known. Lastly, the third setting states that time series of different classes should not be aligned, as they do not contribute to the within variance of each class.

Similarly, the between variance (i.e. the variance between classes) can be defined as follows.

Definition 3: The between variance with an inter-class alignment matrix M is given by:

$$TotBV_M = \frac{1}{nT} \sum_{k=1}^K \sum_{j=1}^p \sum_{l=1}^n \sum_{i=1}^T (x_{ij}^l - A_k(x_{ij}^l + \sum_{k' \neq k} \sum_{l'=1}^{n_{k'}} \sum_{i'=1}^T m_{ii'}^{ll'} x_{i'j}^{l'})^2$$

with

$$A_k = \frac{1}{1 + \sum_{k' \neq k, 1 \leq k, k' \leq K} n_{k'}}$$

and

$$M^{ll'} = \begin{cases} \mathbf{I} & \text{if } l = l' \\ \mathbf{0} & \text{if } y_l = y_{l'} \text{ and } l \neq l' \\ \neq \mathbf{0} & \text{if } y_l \neq y_{l'} \end{cases} \quad (6)$$

where \mathbf{I} and $\mathbf{0}$ are the $(T \times T)$ identity and zero matrices, respectively.

The general form of the alignment matrix M is symmetric wrt to the preceding one, alignments between time series of the same class being forbidden this time, whereas alignments between time series of different classes are taken into account.

As one can note, alignments between time series play a crucial role (through the intra and inter class alignment matrices) in the definition of the within and between variances. To discriminate time series, the question which arises is thus how to learn such matrices so as to be able to minimize the within variance and maximize the between variance. We present an answer to this question in the following section.

III. LEARNING DISCRIMINATIVE ALIGNMENTS

In the following, we mainly focus on the within variance minimization problem, as a similar development can be made for the between variance maximization problem.

The problem of learning the intra-class alignment matrix M to minimize the within variance, i.e. the quantity $TotWV_M$ of Definition 2, can be formulated as a non-convex, quadratic optimization problem with linear inequality and equality constraints:

$$\begin{cases} \arg \min_{m_{ii'}^{ll'}} & TotWV_M = \frac{1}{nT} \sum_{j=1}^p \sum_{l=1}^{n_k} \sum_{i=1}^T (x_{ij}^l - \frac{1}{n_k} \sum_{l'=1}^{n_k} \sum_{i'=1}^T m_{ii'}^{ll'} x_{i'j}^{l'})^2 \\ \text{subject to:} & \forall (i, i', l, l'), m_{ii'}^{ll'} \geq 0, \sum_{i'=1}^T m_{ii'}^{ll'} = 1 \end{cases}$$

The normalization constraints $\sum_{i'=1}^T m_{ii'}^{ll'} = 1$ ensure that for all $S_l, S_{l'}$ each observation i in S_l is connected to at least one observation i' of $S_{l'}$. Furthermore, all the connections are positively weighted.

A standard way to solve the above problem relies on a projected gradient descent, where one iteratively makes a gradient descent on $TotWV_M$ and projects the solution onto the admissible domain provided by the constraints. However, we have no guarantee that the solution after projection has indeed decreased the within variance $TotWV_M$. We introduce here another approach in which the gradient descent

part is replaced by an update of the alignment matrix, which guarantees that the within variance is decreased after projection on the admissible domain, as discussed in section III-B.

A. Learning alignments within and between clusters

The core of the approach we propose to learn the intra-class alignments minimizing the within variance consists in iteratively evaluating the contribution of each linked observations (i, i') to the estimated variance; the weights $m_{ii'}^{ll'}$ are then penalized for links (i, i') increasing the within variance, and this is repeated until the induced within variance stabilizes. This process is illustrated in Algorithm 1, *LearnWAlign*.

Algorithm 1 *LearnWAlign*(X, Y, α)

```

1:  $s = 0$ 
2:  $M^0 \leftarrow$  complete intra-class linkage
3: repeat
4:    $CTot = 0$ 
5:   for all  $(l, l') : y_l = y_{l'} \text{ and } l \neq l' \text{ do}$ 
6:     for all  $(i, i') \in [1, T] \times [1, T] \text{ do}$ 
7:       {evaluation of the within contributions}
8:        $C_{ii'}^{ll'} = TotV_{M^s} - TotV_{M^s \setminus (i, i', l, l')}$ 
9:        $CTot = CTot + |C_{ii'}^{ll'}|$ 
10:    end for
11:  end for
12:   $\mathcal{E} = \{(i, i', l, l') / C_{ii'}^{ll'} > 0\}$ 
13:  repeat
14:    {control of the variance decrease}
15:     $T = True$ 
16:    randomly select  $(i, i', l, l') \in \mathcal{E}$ 
17:    for all  $j = 1, \dots, p \text{ do}$ 
18:      if  $(T \wedge ((\delta_2 \in [0, a]) \vee (\delta_1 \in [0, b]) \vee (-\delta_1 \in [0, b]))) \text{ then}$ 
19:         $T = False$ 
20:         $\mathcal{E} = \mathcal{E} \setminus (i, i', l, l')$ 
21:      end if
22:    end for
23:  until  $T = True$ 
24:  {penalization of the link  $(i, i')$  of  $S_l, S_{l'}$ }
25:   $m_{ii'}^{ll'} = m_{ii'}^{ll'} (1 - \frac{C_{ii'}^{ll'}}{CTot})$ 
26:  {the  $i$ th row normalization}
27:  for all  $t \in [1, T] \text{ do}$ 
28:     $m_{it}^{ll'} = \frac{m_{it}^{ll'}}{1 - m_{ii'}^{ll'} \frac{C_{ii'}^{ll'}}{CTot}}$ 
29:  end for
30:   $s = s + 1$ 
31: until  $\frac{TotV_{M^{s-1}} - TotV_{M^s}}{TotV_{M^{s-1}}} \leq \alpha$  {repeat the learning until the stabilization of the within variance}
32: return( $M^s$ )

```

Starting with an initial intra-class alignment matrix, set to a complete linkage defined by:

$$M^{ll'} = \begin{cases} \mathbf{I} & \text{if } l = l' \\ \frac{1}{T} \mathbf{U} & \text{if } y_l = y_{l'} \text{ and } l \neq l' \\ \mathbf{0} & \text{if } y_l \neq y_{l'} \end{cases} \quad (7)$$

the algorithm *LearnWAlign* proceeds in two steps. In the first step (line 5 to 11), the contribution of each link (i, i') to the within variance is estimated. For this, we evaluate the induced effect on the within variance when (i, i') is removed from the alignment between S_l and $S_{l'}$, two time series of the same class ($y_l = y_{l'}$). Let us denote $M \setminus (i, i', l, l')$ the within alignment after the link (i, i') has been deleted (hence $m_{ii'}^{ll'} = 0$). The contribution $C_{ii'}^{ll'}$ of (i, i') to the

within variance is:

$$C_{ii'}^{ll'} = TotWV_M - TotWV_{M \setminus (i, i', l, l')} \quad (8)$$

Let \mathcal{E} be the set of all the links (i, i') for which contributions $C_{ii'}^{ll'}$ are positive, that is, the links which contribute to increase the within variance. In the second step (line 24 to 29), the weight $m_{ii'}^{ll'}$ of a randomly selected link $(i, i') \in \mathcal{E}$ is penalized, and the impacted row normalized. Note that the weights $m_{ii'}^{ll'}$ are decreased proportionally to $C_{ii'}^{ll'}$, and that this decrease is small as $C_{ii'}^{ll'} \ll CTot$. Based on the updated alignments, the variance decrease rate is estimated (line 31). While this rate is greater than α , steps 1 and 2 are iterated. The procedure finally returns the best learned intra-class alignments. Figure 1a illustrates the initial complete linkage intra- (red links, within each class) and inter- (blue links, between the classes) class alignments, and Figure 1b the learned alignments, where the thickness of the links correspond to the intensity of the learned weights $m_{ii'}^{ll'}$.

The complete linkage used to initialize the intra-class alignment matrix can be replaced by any other choice compatible with the general form of the intra-class alignment matrix given in Definition 2. The complete linkage ensures here that all possible alignments are considered and does not impose any *a priori* constraints on the type of alignments one should look for. The random selection of the elements in \mathcal{E} can also be replaced by other selection strategies, as the one of the element for which $C_{ii'}^{ll'}$ is maximum. However, if this latter element is interesting wrt to its impact on the variance when the associated link is removed, its systematic choice may lead to suboptimal solutions after renormalization. A good compromise is thus to operate a random selection of links of slightly different contributions, as proposed in Algorithm 1. The quantities δ_1 , δ_2 , a and b serve to control (lines 13 to 23) the variance decrease. Their definition and rationale are given in section III-B.

A similar algorithm, *LearnBAlign*, is applied to learn the inter-class alignment matrix maximizing the between variance. As before, the initial alignment is set to a complete linkage:

$$M^{ll'} = \begin{cases} \mathbf{I} & \text{if } l = l' \\ \mathbf{0} & \text{if } y_l = y_{l'} \text{ and } l \neq l' \\ \frac{1}{T} \mathbf{U} & \text{if } y_l \neq y_{l'} \end{cases} \quad (9)$$

and the algorithm proceeds in two steps. In the first step, the set \mathcal{E} of all the links (i, i') for which contributions $C_{ii'}^{ll'}$ are negative, that is, the links which contribute to decrease the between variance, is computed. In the second step, the weight $m_{ii'}^{ll'}$ of a randomly selected link $(i, i') \in \mathcal{E}$ is penalized with $m_{ii'}^{ll'} = m_{ii'}^{ll'}(1 + \frac{C_{ii'}^{ll'}}{CTot})$, and the impacted row normalized. While the increase rate is greater than α , steps 1 and 2 are repeated. The procedure finally returns the best learned inter-class alignments.

Algorithm 2 *LearnBAlign*(X, Y, α)

```

1:  $s = 0$ 
2:  $M^0 \leftarrow$  complete inter-class linkage
3: repeat
4:    $CTot = 0$ 
5:   for all  $(l, l') : y_l \neq y_{l'}$  do
6:     for all  $(i, i') \in [1, T] \times [1, T]$  do
7:       {the evaluation of the between contributions}
8:        $C_{ii'}^{ll'} = TotV_{M^s} - TotV_{M^s \setminus (i, i', l, l')}$ 
9:        $CTot = CTot + |C_{ii'}^{ll'}|$ 
10:    end for
11:  end for
12:   $\mathcal{E} = \{(i, i', l, l') \mid C_{ii'}^{ll'} < 0\}$ 
13:  repeat
14:    {control of the variance increase}
15:     $T = True$ 
16:    randomly select  $(i, i', l, l') \in \mathcal{E}$ 
17:    for all  $j = 1, \dots, p$  do
18:      if  $(T \wedge ((\delta_2 \in [0, a]) \vee (\delta_1 \in [0, b]) \vee (-\delta_1 \in [0, b])))$  then
19:         $T = False$ 
20:         $\mathcal{E} = \mathcal{E} \setminus (i, i', l, l')$ 
21:      end if
22:    end for
23:  until  $T = True$ 
24:  {penalization of the link  $(i, i')$  of  $S_l, S_{l'}$ }
25:   $m_{ii'}^{ll'} = m_{ii'}^{ll'}(1 + \frac{C_{ii'}^{ll'}}{CTot})$ 
26:  {the  $i$ th row normalization}
27:  for all  $t \in [1, T]$  do
28:     $m_{it}^{ll'} = \frac{m_{it}^{ll'}}{1 + m_{ii'}^{ll'} \frac{C_{ii'}^{ll'}}{CTot}}$ 
29:  end for
30:   $s = s + 1$ 
31: until  $\frac{TotV_{M^s} - TotV_{M^{s-1}}}{TotV_{M^{s-1}}} \leq \alpha$  {repeat the learning until the stabilization of the between
    variance}
32: return  $(M^s)$ 

```

B. Convergence of the learning alignment process

We focus here on the algorithm *LearnWAlign*, the development for the algorithm *LearnBAlign* being similar.

As the variance is positive, the convergence of the algorithm is based on the variance decrease throughout the learning process. We study the induced effect on the variance of the penalization of a chosen weight $m_{ii'}^{ll'}$ (hence with fixed (i, i', l, l')). Let us consider the variance as a function V , such that $V(m_{ii'}^{ll'}) = V_M$. We prove that a weak penalization of a link with positive contribution leads to a decrease of the variance. For that, we define a function f as follows:

$$f : \beta \mapsto V_{m_{ii'}^{ll'}} - V(\beta m_{ii'}^{ll'})$$

with $\beta \in [0, 1]$ and $V(\beta m_{ii'}^{ll'})$ denoting the total variance obtained with the link (i, i', l, l') penalized by β and all the links $(i, i'', l, l'')(i'' = 1, \dots, T)$ renormalized to satisfy the constraints. f is thus such that $f(1) = 0$ and $f(0) = C_{ii'}^{ll'}$, this latter quantity being positive for all links in \mathcal{E} .

Let us furthermore introduce the following functions, for

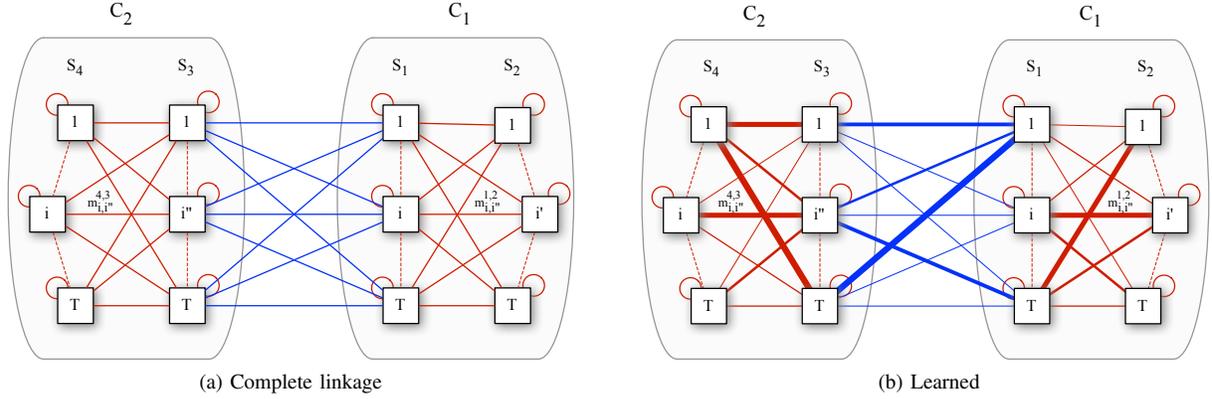


Figure 1: Initial complete linkage (a) and learned (b) intra- and inter-class alignments

given (i, i', j, l, l') :

$$\begin{aligned} \delta_1 &= x_{ij}^l - \sum_{r=1}^M \sum_{t=1}^T m_{it}^{lr} x_{tj}^r \\ \delta_2 &= x_{i'j}^{l'} - \sum_{r=1}^M \sum_{t=1}^T m_{it}^{lr} x_{tj}^r \\ a &= m_{ii'}^{l'l'} x_{ij}^{l'} \\ b &= \frac{m_{ii'}^{l'l'}}{2(1 - m_{ii'}^{l'l'})} \left(1 + \frac{2m_{ii'}^{l'l'} x_{ij}^{l'}}{\delta_1(x_{ij}^l, x_{i'j}^{l'})} \right) \end{aligned}$$

The following property provides conditions for the sign of $f'(1)$ to be different from the sign of $f(0)$, and hence, for links in \mathcal{E} , conditions under which the variance is decreasing when small penalizations are considered.

Property 4: Let Λ be defined by $\delta_1 \times \delta_2$. We have:

- 1) $\text{sign}(-f'(1)) \neq \text{sign}(\Lambda) \Leftrightarrow \delta_2 \in [0, a]$
- 2) $\text{sign}(f(0)) \neq \text{sign}(\Lambda) \Leftrightarrow \delta_1 \in [0, b]$ or $-\delta_1 \in [0, -b]$

Proof (sketch): The proof is based on the consideration of a function g which behaves like f but has a simpler form and is thus easier to manipulate. Indeed, let us consider the function g defined by:

$$\begin{aligned} g(\beta) &= (1 - \beta)(\delta_2 - \beta m_{ii'}^{l'l'} x_{ij}^{l'})((2 - (1 - \beta))\delta_1 \\ &\quad - (1 - \beta)m_{ii'}^{l'l'}(x_{ij}^l + x_{i'j}^{l'} + (\beta + 2)m_{ii'}^{l'l'} x_{ij}^{l'})) \end{aligned}$$

Then f and g are of the same sign, and $\text{sign}(f'(1)) = \text{sign}(g'(1))$. Comparing the sign of both $g(0)$ and $g'(1)$ with the sign of Λ yields the desired result. \square

For all links in \mathcal{E} , $f(0)$ is positive and the variance will decrease for small penalizations ($f'(1) < 0$) as soon as $f(0)$ and $-f'(1)$ have the same sign, which is the sign of Λ . The above property gives us the conditions under which this

happens. The first case in Property 4 occurs when $x_{i'j}^{l'}$ is too close from the neighbors of x_{ij}^l , so that the renormalization of the weights after the penalization would induce more effects than the penalization itself, and would increase the variance. The second case in Property 4 appears when the distance between $x_{i'j}^{l'}$ and its neighbors is low, so that the variation of the weight of $x_{i'j}^{l'}$ moves x_{ij}^l and the mean of its neighbors away. In that case, we are close to the convergence of the algorithm. Should these two cases occur simultaneously, we would observe a decrease of the variance. However, we decide to exclude this case in the algorithm, granted this seldom occurs and thus uselessly complexifies the algorithm.

Lastly, in practice, the conditions expressed in Property 4 and used in algorithm *LearnWAlign* to control the variance decrease are usually satisfied and only need be checked when one is close to convergence.

C. Complexity of the learning process

We again focus here on the learning of intra-class alignments, the development for the inter-class alignments being similar. For a given class k , the quantity $C_{ii'}^{l'l'}$ in *LearnWAlign* can be rewritten as:

$$\begin{aligned} \frac{1}{nT} \sum_{j=1}^p (x_{ij}^l - \frac{1}{n_k} \sum_{l_2=1}^{n_k} \sum_{i_2=1}^T x_{i_2j}^{l_2} m(b)_{ii_2}^{ll_2})^2 \\ - (x_{ij}^l - \frac{1}{n_k} \sum_{l_2=1}^{n_k} \sum_{i_2=1}^T x_{i_2j}^{l_2} m(a)_{ii_2}^{ll_2})^2 \end{aligned}$$

where $m(b)_{ii_2}^{ll_2}$ and $m(a)_{ii_2}^{ll_2}$ respectively denote the weight of link (i, i_2) , between the two series l and l_2 , before and after the link (i, i') has been removed. Hence, the complexity for computing $C_{ii'}^{l'l'}$ for all (i, i', l, l') amounts to $O(pn_k^3 T^3)$, which is the dominating term in *LearnWAlign*.

As we mentioned earlier, the problem solved by *LearnWAlign* can be formulated as a non-convex, quadratic optimization problem with linear inequality and equality con-

straints. Computing the Lagrangian of this problem leads to a system of linear equations which requires, in the worst case, $O(\sum_{k=1}^K n_k^6 T^6)$ operations to be solved. Furthermore, the solutions of this system are extrema, and thus need to be checked for minimality. For the projected gradient approach, where one iteratively makes a gradient descent on $TotV_M$ and projects the solutions onto the admissible domain provided by the constraints, the term dominating the complexity of the algorithm is associated to the computation of the gradient, and has a complexity of $O(p n_k^3 T^3)$. Thus, both *LearnWAlign* and the projected gradient descent approach have the same overall complexity (up to a factor controlling the number of iterations). There are however two main advantages in using *LearnWAlign* and not a projected gradient descent here: (i) We have no guarantee that the solution after projection has indeed decreased $TotWV_M(k)$ in the projected gradient descent approach, whereas we have such a guarantee for *LearnWAlign*; (ii) *LearnWAlign* does not require the setting of an additional parameter controlling the descent along the gradient: as soon as $C_{ii'}^{ll'} > 0$, one knows that the variance can be minimized and that the update in *LearnWAlign* will provide a valid solution.

IV. TIME SERIES CLASSIFICATION BASED ON THE LEARNED ALIGNMENTS

We present here the use of the learned alignments in k -nearest neighbors classification. To do so, we introduce a distance which simultaneously takes into consideration the similarities within classes and the dissimilarities between classes specified by the intra- and inter-class alignments. To evaluate the distance between a new time series S_{test} and a sample series S_l , one needs first to align S_{test} to S_l . For this, we rely on an optimistic hypothesis which consists in assuming that S_l is a good approximation for S_{test} and thus that the alignment between the two is similar to the one between S_l and its nearest neighbor in class y_l . However, by doing so, one relies only on the intra-class alignments learned in the previous section, and ignores the information brought by inter-class alignments. To take this latter information into account, we propose to weight variables according to their dispersion within class y_l and within the class containing the nearest impostor of S_l (the impostor of any training example is the nearest neighbor of that example which does not belong to the same class). The following development formalizes these ideas.

Let us define the distance between two training samples $S_l, S_{l'}$ based on their intra- or inter-class alignments as:

$$d_M^2(S_l, S_{l'}) = \sum_{i, i'} \left(\frac{m_{ii'}^l + m_{ii'}^{l'}}{2} \right) \sum_{j=1}^p (x_{ij}^l - x_{i'j}^{l'})^2$$

where M alternatively denotes the learned intra-class alignment for $y_l = y_{l'}$, and inter-class for $y_l \neq y_{l'}$. In addition, we denote by $c_l \neq y_l$ the class of the impostor of the time

series S_l :

$$c_l = \arg \min_{k \neq y_l} \left(\sum_{l', y_{l'}=k} d_M^2(S_l, S_{l'}) \right)$$

For each series S_l , one can distinguish, on the one hand, the set of links connecting S_l to the series of the same class and, on the other hand, the set of links connecting S_l to the series of its nearest class c_l . Both sets of links induce two standard deviations evaluating the dispersion of its linked values within y_l and within c_l . This dispersion can be summarized through the following function:

$$\sigma(k, i, j, y_l) = \sum_{l', y_{l'}=k} \sum_{i'=1}^T m_{ii'}^{ll'} (x_{i'j}^{l'}) - \sum_{l', y_{l'}=k} \sum_{i'=1}^T m_{ii'}^{ll'} (x_{i'j}^{l'})^2$$

The quantities of interest to us here are $\sigma(y_l, i, j, y_l)$, for the standard deviation within the same class, and $\sigma(c_l, i, j, y_l)$ for the standard deviation with the class of the impostor. These estimated standard deviations can be used to give more weight to variables ensuring both homogeneity within classes and a good differentiation from the impostor class, namely providing lower values for $\sigma(y_l, i, j, y_l)$ and higher for $\sigma(c_l, i, j, y_l)$. To do so, we propose the following distance between a new test series S_{test} and a training sample S_l , relying on the closest neighbor S_r of S_l within class y_l :

$$r = \arg \min_{l', y_{l'}=y_l} (d_M^2(S_l, S_{l'}))$$

The final distance is defined by:

$$D(S_{test}, S_l) = \sum_{i, i'} \left(\frac{m_{ii'}^l + m_{ii'}^r}{2} \right) \sum_{j=1}^p \frac{\sigma(c_l, i, j, y_l)}{\sigma(y_l, i, j, y_l)} (x_{i'j}^{test} - x_{i'j}^l)^2 \quad (10)$$

V. APPLICATION AND COMPARISON STUDY

To assess the effectiveness of the learned discriminative alignments, three public time series datasets are considered: Cylinder-Bell-Funnel (CBF) [13], Synthetic Control Chart (CC), and Character Trajectories (TRAJ) [14], also available at [15] (www.cs.ucr.edu/~eamonn/time_series_data/). Note that, for all these data, time series within the same class share a single, common global structure. To highlight the additive value of the proposed approach on more complex temporal peculiarities, we propose two new datasets, downloadable at (blind review), where time series within a same class may have distinctive global behaviors while sharing common local features. The first dataset BME is composed of three classes *Begin*, *Middle*, and *End* (Figure 2). In the *Begin* class, the time series are characterized by a little bell that arises during the Begin period. They may show the same global behavior or not, depending on the top or bottom position of the right-side large bell. The *Middle* class is composed of time series characterized by a centered large bell. In the *End* class, time series are described by a bell arising at the

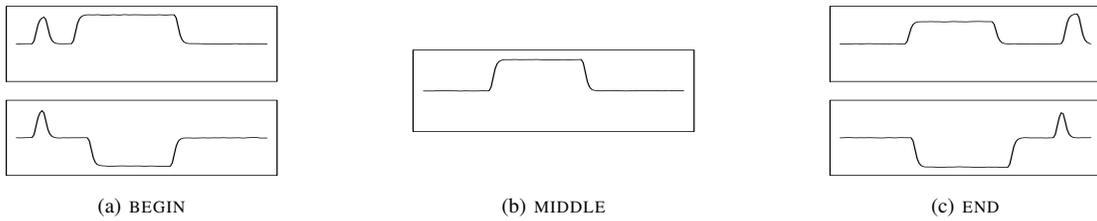


Figure 2: The main behaviors within BME classes: BEGIN, MIDDLE, and END

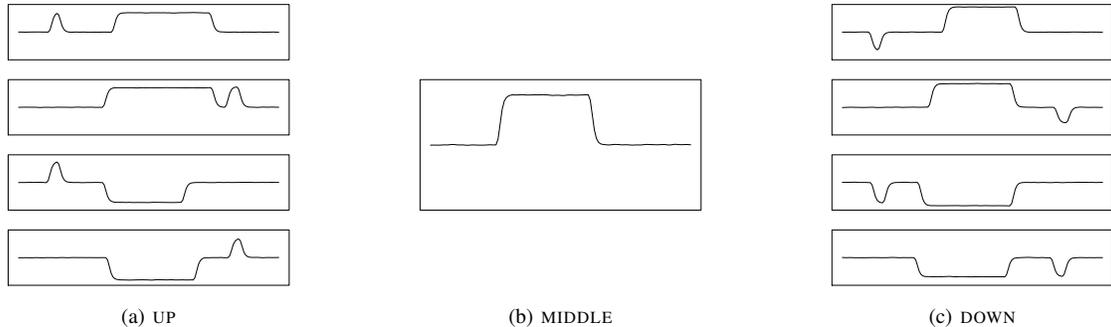


Figure 3: The main behaviors within UMD classes: UP, MIDDLE, and DOWN

end period and similarly may show the same global behavior or not, depending on the top or bottom position of this left-side large bell. The second dataset UMD, composed similarly of three classes *Up*, *Middle*, and *Down*, introduces more complexity since the local shared features may occur at any time stamps as illustrated in Figure 3. Table I provides the main characteristics of the five considered datasets, including the number of time series (Size), the number of classes (Cla.Nb), the number of time series per class (TS.Cla), the length of the time series (Leng), and whether the time series are univariate or multivariate (Multi), real or synthetic (Real).

Table I: Datasets description

Dataset	Size	Cla.NB	TS.Cla	Leng	Mult	Real
CBF	300	3	100	128	No	No
CC	600	6	100	60	No	No
TRAJ	1000	20	50	20	Yes	Yes
BME	300	3	100	128	No	No
UMD	300	3	100	150	No	No

The algorithms *LearnWAlign* and *LearnBAlign* are applied on the five datasets. Figure 4 illustrates the learned links and their weights between two *Cylinder* time series. Figure 4 (left) visualizes the links between one observation of S_l and all observations of S_r . The bold lines, which connect the observation of S_l to the plateau of S_r , highlight the highly weighted links (i.e., higher values of $m_{ii'}^{ll'}$), whereas dotted lines correspond to unconnected observations

(i.e., $m_{ii'}^{ll'} \approx 0$). Figure 4 provides the learned intra-class alignment $M^{ll'}$ between S_l and S_r . Light cells identify the highly connected regions (i.e., highly weighted links). For instance, the central light square references a strong linkage between the plateaus of the two *Cylinder* class time series.

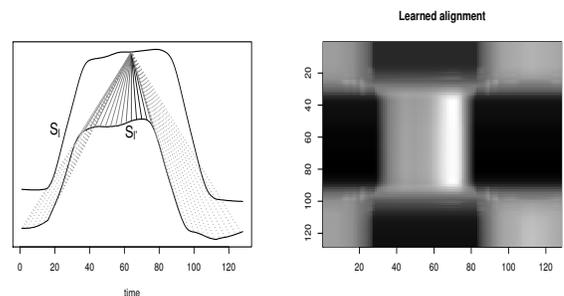


Figure 4: The learned intra-class alignment between two *Cylinder* time series of CBF dataset

The aims of the following sections are, on the one hand, to evaluate the relevance of the learned alignments to discriminate time series classes. For this, a conventional within/between variance criterion is considered to measure the clusters homogeneity and separability. On the other hand, to estimate the performances of the defined discriminative distance to classify time series, and to compare it with the standard euclidean and dynamic time warping distances.

The within/between variance criterion is a conventional index in discriminant analysis; the lower the variance within clusters, and the higher the variance between clusters, the better is the clusters discrimination. For each dataset, Figures 5 and 6 show the progression of the within and between variances during the learning process. Particularly, they indicate the decrease (resp. increase) function of the within (resp. between) variances V_{M^s} and provide a comparison with the within and between variances V_{M^I} , $V_{M^{DTW}}$ and V_{M^0} based on the standard euclidean, the dynamic time warping and the complete linkages (denoted M^0), respectively. The red star indicates the optimal learned within and between variances. For a stopping threshold of $\alpha = 10^{-3}$, the intra and inter class alignments processes converges in less than 50 and 80 iterations respectively. Finally, Table II gives, for the five datasets, the optimal within and between variances, and the obtained within/between discrimination ratios.

Figure 7 gives an example of the learned intra-class alignment between two time series of the class *Down* (UMD dataset). It is interesting to note that little bells (shared event within the class), occurring at the beginning for the first time series and at the last period for the second one, are highly weighted; whereas the large bells are weakly linked, although similar for the considered time series.

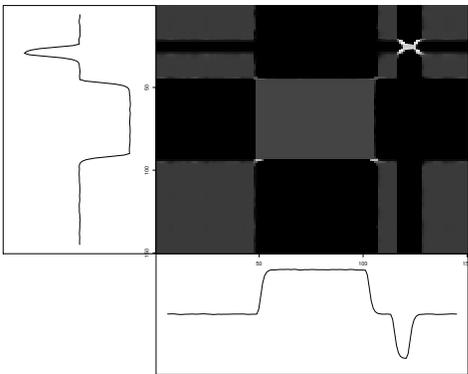


Figure 7: The learned intra-class alignment between two *Down* time series of UMD dataset

The 1-nearest neighbor classification based on the discriminative distance D (Eq. 10) is performed on the five studied time series datasets through a one hold-out protocol. Table III compares the misclassification error rates with those obtained by the standard euclidean dE and dynamic time warping DTW distances.

VI. DISCUSSION AND PERSPECTIVES

Let us first discuss and analyze the behavior of the proposed learning process by considering Figures 5 and 6. The clearly monotonically decreasing behavior of the

Table III: Nearest neighbor classification error rates

	CBF	CC	TRAJ	BME	UMD
D	10.50	1.53	1.20	4.00	9.72
dE	10.47	4.70	1.20	17.70	16.96
DTW	33.20	9.63	1.90	13.40	13.29

within variances (versus the increasing behavior of the between variances) throughout all datasets points out the pertinence of the links weights penalization to maximize cluster cohesion and isolation. The drastic progression of the variances at the beginning and its slowdown until ending at a plateau stress two characteristics; namely, the convergence of the proposed learning process and its efficiency in speeding up convergence at the beginning of the process, when the objective function is far from the optimal.

We can note from Figure 5, that the optimal learned within variances are significantly lower than V_{M^I} and slightly equivalent to $V_{M^{DTW}}$ for all datasets. The within variances V_M , V_{M^I} and $V_{M^{DTW}}$ are closer for TRAJ data; this indicates that both the dynamic time warping alignments M^{DTW} and the learned ones M lead to the identity linkage. Indeed, in Figure 5, we can see that the learned alignments within, for instance, the clusters "c", "o", "l", "e", and "u" strongly reveal an identity linkage. For Figure 6, we note the continuous increase of the between variances during the learning process, with optimal between variances V_M significantly higher than V_{M^I} and $V_{M^{DTW}}$.

Table II shows a better discrimination (i.e., higher cohesion and isolation) for the learned alignments with the lowest discrimination ratios ρ for all datasets. The alignments considered by the dynamic time warping M^{DTW} and the learned ones M lead to better clusters cohesion than the euclidean alignments M^I , however the learned alignments ensure better clusters isolation than the standard alignments M^{DTW} and M^I .

Through the misclassification error rates summarized in Table III, we can see that the discriminative distances provide a good improvement of the performances for the CBF and CC datasets, and lead to quite similar performances on the TRAJ dataset, since as discussed above the learned alignments converges to the euclidean alignments (Figure 5). However, discriminative distances outperform significantly the standard euclidean and dynamic time warping for the more complex BME UMD datasets (i.e. classes are composed of time series of distinctive global behaviors).

In the framework of time series discrimination, it may be restrictive to determine temporal alignments for each couple of time series regardless of the dynamic of the remaining time series. The proposed temporal alignment strategy moves one step forward to address this issue and to take more into consideration the time series structure within and between clusters. Although *LearnWAlign* allows us to

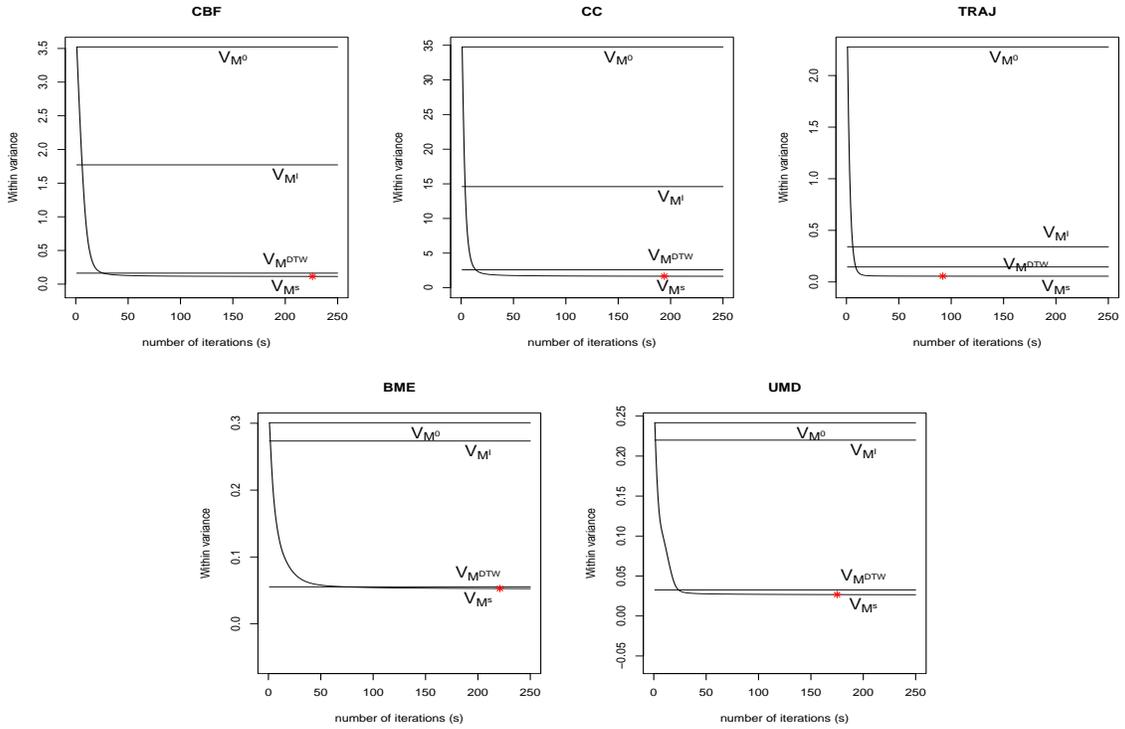


Figure 5: The Within-variance progression during the learning process ($\alpha = 10^{-3}$).

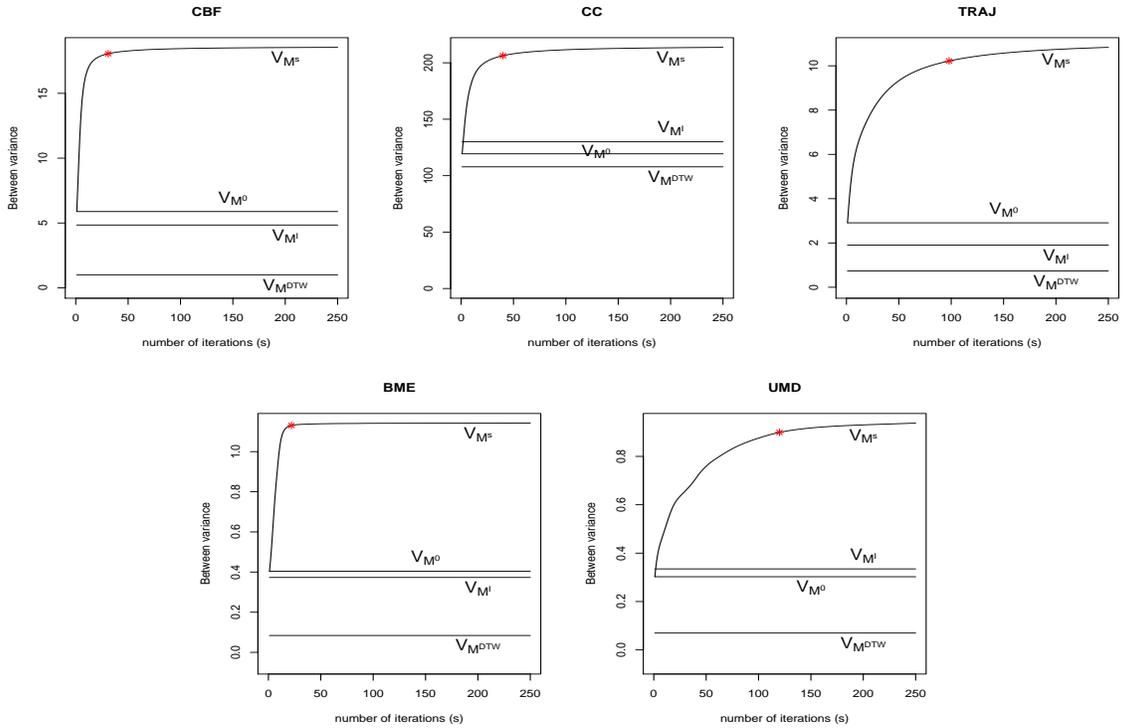


Figure 6: The Between-variance progression during the learning process ($\alpha = 10^{-3}$).

Table II: The evaluation of the time series discrimination based on the within/between variance ratio

	Within			Between			$\frac{Within}{Between}$		
	V_M	V_{MI}	V_{MDTW}	V_M	V_{MI}	V_{MDTW}	ρ	ρ^I	ρ^{DTW}
CBF	0.119	1.771	0.163	18.441	4.844	1.004	0.006	0.366	0.162
CC	1.732	14.597	2.587	212.339	130.001	107.818	0.008	0.112	0.024
TRAJ	0.05	0.341	0.145	10.830	1.902	0.739	0.005	0.305	0.196
BME	22.161	65.955	22.734	199.476	109.089	35.548	0.111	0.605	0.640
UMD	0.027	0.220	0.033	0.899	0.335	0.070	0.030	0.657	0.469

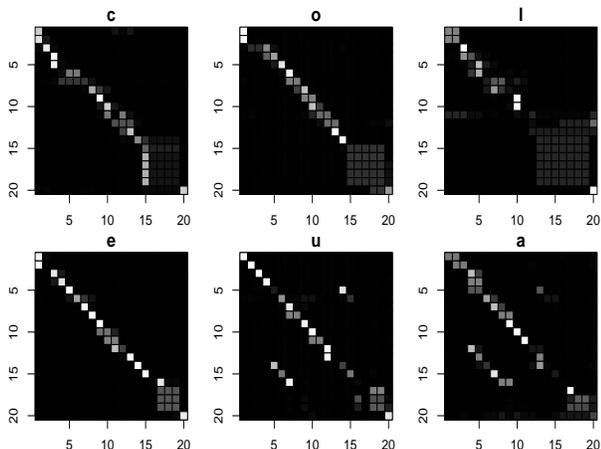


Figure 8: The learned alignments M within the six clusters (“c”, “o”, “l”, “e”, “u”, “a”) of TRAJ dataset

align time series of the same class while considering the structure of the other time series within the class, it ignores how time series of the remaining clusters are structured.

On the other hand, its between version aligns time series of different classes by considering the structure of all time series across different classes, but it still ignores which time series are of the same class. Accordingly, our aim in a future study consists in merging these two algorithms so that the learning process simultaneously includes the entire time series structure within and between clusters.

VII. CONCLUSION

We have proposed in this paper a new temporal alignment algorithm minimizing an intra-class variance and maximizing an inter-class variance. The main idea behind this algorithm is based on strengthening or weakening links according to their contribution to the variability within and between classes. To do so, we have extended the standard definition of the variance to a set of time series, and then to a partition of a set of time series. We have furthermore introduced a new distance between time series which exploits the learned alignments. We have then used this distance for k -nearest neighbor classification on real and synthetic datasets. The results of our experiments show that the approach we

have developed is able to capture fine-grained distinctions between time series across classes, and outperforms standard approaches on several different datasets, all the more so that the correspondence between time series within the same class is complex.

REFERENCES

- [1] J. Kruskal and M. Liberman, *The symmetric time warping algorithm: From continuous to discrete*. In *Time Warps, String Edits and Macromolecules*. Addison-Wesley., 1983.
- [2] G. Navarro, “A guided tour to approximate string matching,” *ACM Computing Surveys*, vol. 33, no. 1, pp. 31–88, 2001.
- [3] D. Sankoff and J. Kruskal, *Time warps, string edits, and macromolecules: the theory and practice of sequence comparison*. Addison-Wesley, 1983.
- [4] Y. Shou, N. Mamoulis, and D. W. Cheung, “Fast and exact warping of time series using adaptive segmental approximations,” *Machine Learning Journal*, vol. 58, no. 2-3, pp. 231–267, 2005.
- [5] X. Xi, E. Keogh, C. Shelton, and L. Wei, “Fast time series classification using numerosity reduction,” in *Proc. of the International Conference on Machine Learning*, 2006, pp. 1033–1044.
- [6] S. J. Gaffney and P. Smyth, “Joint probabilistic curve clustering and alignment,” *In Advances in Neural Information Processing Systems*, vol. 17, pp. 473–480, 2005.
- [7] J. Listgarten, R. Neal, S. Roweis, R. Puckrin, and S. Cutler, “Bayesian detection of infrequent differences in sets of time series with shared structure,” *Neural Information Processing Systems*, vol. 19, no. 905–912, 2007.
- [8] J. Ramsay and X. Li, “Curve registration,” *Journal of the Royal Statistical Society*, vol. B, no. 60, pp. 351–363, 1998.
- [9] W. D., “Multivariate spatial correlation: A method for exploratory geographical analysis,” *Geographical Analysis*, vol. 17, no. 4, pp. 263–283, 1985.
- [10] T. A. Banet and L. Lebart, “Local and partial principal component analysis and correspondence analysis,” *Computational Statistics*, pp. 113–118, 1984.
- [11] T. A. Banet, “Local and partial correspondence analysis: application to the analysis of electoral data,” *Computational statistics quarterly*, vol. 2, pp. 89–103, 1988.

- [12] J. Thioulouse, D. Chessel, and S. Champely, "Multivariate analysis of spatial patterns: a unified approach to local and global structures," *Environmental and Ecological Statistics*, vol. 2, pp. 1–14, 1995.
- [13] N. Saito, "Local feature extraction and its application using a library of bases," Ph.D. dissertation, Department of Mathematics, Yale University., 1994.
- [14] A. Asuncion and D. Newman, "Uci machine learning repository [<http://www.ics.uci.edu/mllearn/mlrepository.html>]. irvine, ca: University of california, school of information and computer science." 2007.
- [15] E. Keogh, X. Xi, L. Wei, and C. A. Ratanamahatana, "The ucr time series classification/clustering homepage," 2006.

