Nature-inspired Approaches for Distance Metric Learning in Multivariate Time Series Classification

Izaskun Oregi*, Javier Del Ser*,†,‡, Aritz Pérez‡ and José A. Lozano†,‡
*TECNALIA, 48160 Zamudio, Bizkaia, Spain
Email: {izaskun.oregui, javier.delser}@tecnalia.com
†University of the Basque Country UPV/EHU, Bilbao, Bizkaia, Spain
Email: {javier.delser, ja.lozano}@ehu.eus
‡Basque Center for Applied Mathematics BCAM, 48009 Bilbao, Bizkaia, Spain
Email: aperez@bcamath.org

***Abstract*—The applicability of time series data mining in many different fields has motivated the scientific community to focus on the development of new methods towards improving the performance of the classifiers over this particular class of data. In this context the related literature has extensively shown that dynamic time warping is the similarity measure of choice when univariate time series are considered. However, possible statistical coupling among different dimensions make the generalization of this metric to the multivariate case all but obvious. This has ignited the interest of the community in new distance definitions capable of capturing such inter-dimension dependences. In this paper we propose a simple dynamic time warping based distance that finds the best weighted combination between the dependent – where multivariate time series are treated as whole – and independent approaches – where multivariate time series are just a collection of unrelated univariate time series – of the time series to be classified. A benchmark of four heuristic wrappers, namely, simulated annealing, particle swarm optimization, estimation of distribution algorithms and genetic algorithms are used to evolve the set of weighting coefficients towards maximizing the cross-validated predictive score of the classifiers. In this context one of the most recurring classifiers is nearest neighbor. This classifier is couple with a distance that as afore mentioned, in most cases, have been dynamic time warping. The performance of the proposed approach is validated over datasets widely utilized in the related literature, from which it is concluded that the obtained performance gains can be enlarged by properly decoupling the influence of each dimension in the definition of the dependent dynamic time warping distance.**

I. Introduction

Time series, conceived as a list of data points sorted in time order, are present in many different fields such as telecommunications, finance and biomedicine, among others [1], [2], [3]. In such areas it is often the case that time series are assigned a category or label (e.g. the chance of a customer to churn from a telecommunications company based on the record of transactions), which is of interest for the underlying application (e.g. customer retention).

In order to predict the label associated to new time series, supervised learning aims at building classification models based on a record of past labeled time series. The most common time series classification method is the k-nearest neighbour (k-NN) scheme [4]: when this model is queried for the label of a new item to be predicted, the distance to each sample in the training set is computed, from which the predicted label results as the majority class among the labels of the k closest training examples. Thorough reviews on time series classification algorithms are presented in [5], [6]. In parallel to the more traditional approach to build these models based on the extraction of features from the time series, a research trend of vibrant activity in the literature gravitates on the use of tailored distances between time series and their exploitation in learning models that rely on pairwise similarity measures. Hence to compute the distance between two time series, not only feature-based similarity measures can be used but also model and raw data-based distances [7]. Model- and feature-based approaches assume a priori knowledge on the properties of the sources that generated the time series. In this paper we will focus on raw distance measures, which override any assumption on the characteristics of the time series.

Multiple studies have shown that among all raw data-based similarity measures, the so-called Dynamic Time Warping (DTW) is the most competitive approach for time series classification [8]. In essence DTW is an elastic measure of similarity capable of stretching and/or shrinking time series along time prior to their distance computation in order to accommodate local time shifts and warps. Hence, DTW computes the minimum distance between two time series by aligning the coordinates of the points comprising both sequences. Mathematically speaking, consider two sequences $\mathbf{t} = (t_1, t_2, \ldots, t_N)$ and $\mathbf{u} = (u_1, u_2, \ldots, u_M)$, and a $N \times M$ grid where each coordinate pair (i, j) ($i \in \{1, \ldots, N\}$ and $j \in \{1, \ldots, M\}$) is assigned a value equal to the distance $d(t_i, u_j) = (t_i - u_j)^2$. To compute the optimal alignment between $\mathbf{t}$ and $\mathbf{u}$, DTW finds a *warping* path $\mathbf{w}$ from $(1, 1)$ to (N, M) through the grid that minimizes its total cumulative weight. Let this path be denoted as $\mathbf{w}^* = (w_1, \ldots, w_l, \ldots, w_L)$, with $w_l = d(t_{i(l)}, u_{j(l)})$ and $(i(l), j(l))$ being the coordinates of the l-th step of the warping path through the grid. The DTW distance is given by

$$\mathrm{DTW}(\mathbf{t}, \mathbf{u}) = \min_{\mathbf{w}} \sqrt{\textstyle\sum_{l=1}^{L} w_l} \tag{1}$$

$$\text{subject to} \quad (i(1), j(1)) = (1, 1), \tag{2}$$

$$\Delta(i, j) \in \{(1, 1), (0, 1), (1, 0)\}, \tag{3}$$

$$(i(L), j(L)) = (N, M), \tag{4}$$

where $\Delta(i, j) = (i(l) - i(l-1), j(l) - j(l-1))$ for $l \geq 2$. Since

978-1-5090-4601-0/17/$31.00

the number of allowed paths increases exponentially with time series length, dynamic programming is used for the search of the minimum distance (path). This way, $\text{DTW}(\mathbf{t},\mathbf{u}) = \sqrt{\gamma(N,M)}$ where $\gamma(N,M)$ is the minimum cumulative distance in the final cell and

$$\gamma(i,j)=(t_i\text{-}u_j)^2 + \min\{\gamma(i\text{-}1,j\text{-}1),\gamma(i\text{-}1,j),\gamma(i,j\text{-}1)\} \quad (5)$$

with $\gamma(i,0)=\gamma(0,j)=\infty$, $\gamma(0,0)=0$. It is important to bear in mind the computational cost of $\text{DTW}(\mathbf{t},\mathbf{u})$; in particular, the complexity using the recurrence in (5) is $\mathcal{O}(NM)$.

When the constituent points of a time series have several dimensions we deal with multivariate time series (MTS). A straightforward approach to account for the multidimensionality of MTS in the above distance definition could be to treat dimensions separately (independently) or jointly (dependently). The first approach assumes that all features of the time series under comparison are independent, yielding a measure of warping distance computed as the sum of the DTW for each dimension, i.e.

$$\text{DTW}_I(\mathbf{T},\mathbf{U}) \doteq \frac{1}{D}\sum_{d=1}^{D}\text{DTW}(\mathbf{t}^{(d)},\mathbf{u}^{(d)}), \quad (6)$$

where $\mathbf{t}^{(d)} = (t_1^d,\ldots,t_N^d)$ denotes the N-length time series corresponding to the d-th dimension of multivariate sequence $\mathbf{T} = (\mathbf{t}^{(1)},\ldots,\mathbf{t}^{(d)},\ldots,\mathbf{t}^{(D)})$. However, this independence does not necessarily hold in most practical applications, as there may appear relationships between different dimensions due to latent variables. The dependent distance measure, $\text{DTW}_D(\mathbf{T},\mathbf{U})$, is the extension of the pairwise distance metric in the $N\times M$ grid using, as the inner pairwise distance metric, the multivariate version of the squared Euclidean distance considering all dimensions, i.e.

$$d(\mathbf{t}_i,\mathbf{u}_j) = \sum_{d=1}^{D}(t_i^d - u_j^d)^2 \quad (7)$$

where $\mathbf{t}_i = (t_i^1,\ldots,t_i^d,\ldots,t_i^D)$.

A. Related Work and Contribution

The design of a distance that properly captures and exploits the exiting interrelationships between dimensions remains an active research area in the field of multivariate time series classification. Bankó and Abonyi in [9] presented a classification algorithm that combines DTW and PCA-based segmentation in a hybrid scheme coined as *correlation based dynamic time warping* (CBDTW). The classification algorithm segments MTS homogeneously using, as the segmentation cost function, the Hotellings T^2 statistic (i.e. the MTS distance to the origin in the principal components space) or the Q reconstruction error, which represent the information loss between the original data and its projection in the space of principal components. Once the segmentation is done, the DTW is computed. Likewise, [10] proposes a simple algorithm which selects the DTW measure – either DTW_I or DTW_D – that scores best when predicting the labels, using k-NN, of the time series dataset under analysis. Interestingly for the scope of this manuscript, the authors [10] discovered that the threshold to select one distance or another depends on the training data used for its calculation, which ultimately unveils that practical databases feature a mixture of independence and dependence relationships between their dimensions that should be exploited in the definition of the distance between series. Recently, Mei et al. in [11] propose a similarity measure called *Mahalanobis Distance based DTW* (MDDTW) that combines Mahalanobis Distance learning and DTW for classification tasks. In the proposed method, DTW_D is utilized to find the distance between MTS. However, instead of using Expression (7) as the pairwise distance, they use the generalized Mahalanobis distance given by

$$d(\mathbf{t}_i,\mathbf{u}_j) = (\mathbf{t}_i - \mathbf{u}_j)\mathbf{M}(\mathbf{t}_i - \mathbf{u}_j)^T, \quad (8)$$

where $\mathbf{M}$ is a symmetric positive semi-definite matrix. As the authors conclude, the model learning is computationally expensive due to the need for computing the DTW distance for different values of $\mathbf{M}$ during the learning process.

Our work is aligned with the above noted need for computationally efficient multivariate distance learning algorithms and recent contributions dealing with parametric distance measures for MTS classification [12]. Specifically, we propose a weighted distance that combines both independent and dependent DTW components in its definition. Since the selection of a proper metric is strongly biased by the dataset at hand, we resort to a heuristic wrapper driven by the validated predictor score. Then, the optimization is just a distance-based learning problem operating on the modified distance space spanned by the weighted combination of DTW metrics. This technical approach is aligned with past contributions dealing with the use of wrapping heuristics for distance-based learners (e.g. [13]), where the distance measure along samples is optimized by weighting the value of their features rather than by tuning the metric itself. In this paper, in order to optimize the weights, four nature-inspired evolutionary meta-heuristics are used, simulated annealing, particle swarm optimization, genetic algorithms and estimated distribution algorithms. The four alternatives are evaluated and compared by computer experiments over datasets utilized in the literature. From the obtained results we will not only show the performance improvements achieved by every heuristic, but also provide an intuitive insight on how further gains could be achieved.

The remainder of the paper is organized as follows: Section II introduces the definition of our DTW-based multivariate distance and the optimization procedure by means of an heuristic wrapper. Section III provides experimental results and finally, Section IV concludes the paper.

II. Proposed Distance Metric Formulation

Specifically our proposal gravitates on reformulating the similarity of two MTS, $\mathbf{T}$ and $\mathbf{U}$ as follows:

$$\text{DTW}_{opt}(\mathbf{T},\mathbf{U}) = \sum_{d=1}^{D}\omega_d\text{DTW}(\mathbf{t}^{(d)},\mathbf{u}^{(d)}) + \left(1-\sum_{d=1}^{D}\omega_d\right)\text{DTW}_D(\mathbf{T},\mathbf{U}), \quad (9)$$

where $0 \leq \omega_d \leq 1/D$ and $0 \leq \sum_{d=1}^{D} \omega_d \leq 1$. Note from the definition that both, DTW_D and DTW_I are computed and storage in advance. Hence, in terms computational cost, the optimization of $\boldsymbol{\omega} = \{\omega_d\}_{d=1}^{D}$ values is more efficient. The above definition allows for the flexibility required to tackle distance-based multidimensional time series classification: depending on the values given to $(\omega_1, \ldots, \omega_D)$, the resulting distance could range from the total dependence assumption ($\omega_d = 0 \ \forall d \in \{1, \ldots, D\}$) to the case where all dimensions are assumed to be independent of each other ($\sum_{d=1}^{D} \omega_d = 1$).

To speed up DTW calculations warping constraints such as the so-called Sakoe-Chiba Band [14] or the Itakura Parallelogram [15] have been extensively used in related works. The use of warping windows introduces more restrictions to the definition of the baseline DTW metric as per (1). Consequently, the warping alignment between time series is constrained to a certain time range and in the multidimensional case, relations and statistical interactions among different dimensions might be altered. Since the difference between completely dependent and independent scenarios should rely on the correlation between different dimensions, no warping constraints will be applied in our DTW computations.

A. Optimization Procedure

As show in Figure 1 values of $\boldsymbol{\omega} = \{w_d\}_{d=1}^{D}$ are optimized by means of a heuristic wrapper, where the score function is the predictive accuracy of 1-NN. Since the predictive accuracy is unknown we need to estimate it from data. Taking into account that 1-NN is very sensitive to changes in the training set and that it can suffer from over fitting, following [16], we estimate the predictive accuracy using an estimator with low variance. In particular we have used, the m-repeated k-fold cross-validation($m \times k$-cv) with $m = 10$ and $k = 2$. The $m \times k$-cv estimator consists of averaging m different performance estimates provided by a stratified k-fold cross-validation (k-cv).

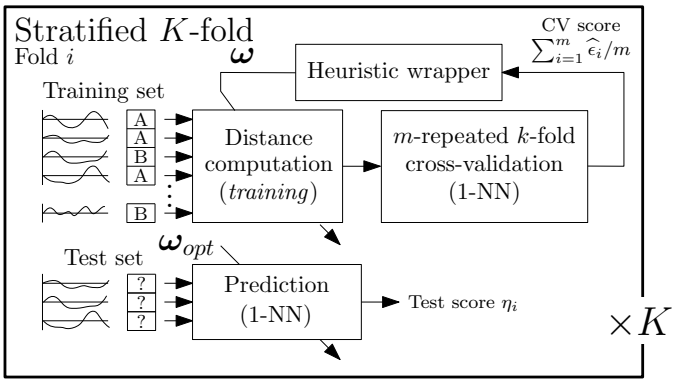

Fig. 1: Schematic diagram of the proposed wrapper scheme for distance metric optimization.

Algorithm 1 sketches the procedure to measure the fitness of a set of $\boldsymbol{\omega}$ candidate values for any given training dataset. The k-cv error estimation procedure splits the training data into mutually exclusive k-folds (line 3), from which training and validation sets are successively constructed (lines 5 and 6). Given a value of $\boldsymbol{\omega}$ the classifier is trained and evaluated using Expression (9) as the distance among examples in the 1-NN model. The k-cv estimation results from averaging the performance scores achieved with each fold.

Algorithm 1 Computation of the fitness for the wrapper

1: **procedure** FITNESS(k, m, $\boldsymbol{\omega}$, training data)
2: **for** $i \in \{1, \ldots, m\}$ **do**
3: Shuffle the training data
4: Sample k stratified folds from the training data
5: **for** $j \in \{1, \ldots, k\}$ **do**
6: Set the j-th fold as the validation set
7: Set the rest of folds as the training set
8: **for all** sample in the validation set **do**
9: Compute $\text{DTW}_{opt}(\cdot, \cdot)$ between the validation sample and every sample in the training set
10: Predict the label for the validation sample to be that of the training sample with $\min \text{DTW}_{opt}$
11: **end for**
12: Compute prediction score ϵ_j by comparing predicted and true labels
13: **end for**
14: Calculate and store k-cv as $\widehat{\epsilon}_i = \sum_{j=1}^{k} \epsilon_j / k$
15: **end for**
16: **return** fitness value given by $\sum_{i=1}^{m} \widehat{\epsilon}_i / m$
17: **end procedure**

Regarding the heuristic wrapper, four are the different solvers utilized to seek the optimum value of weights $\boldsymbol{\omega} = \{w_d\}_{d=1}^{D}$: Simulated Annealing (SA), Particle Swarm Optimization (PSO), Estimated Distribution Algorithms (EDAs) and Genetic Algorithms (GA). A brief description of each of these methods is next provided:

- SA [17] is an iterative low-complexity optimization algorithm known to efficiently tackle problems with small number of variables. The search process underlying this heuristic emulates the annealing technique in metallurgy, by which a material is heated and cooled in a controlled fashion so as to lead it to a state with minimum internal energy and hence, maximum hardness. This search process is controlled mainly by 1) the method to permute the candidate solution at a given iteration; and 2) the temperature T of the material, which sets the probability that the mutated individual is accepted as the candidate solution of the algorithm.
- PSO [18] consists of a swarm of particles moving in the space of candidate solutions. Each individual in the swarm is characterized by its position over the search space (which in turn represents the solution $\boldsymbol{\omega}$ proposed by the particle), a velocity vector $\mathbf{v}$ and the memory of both its own best solution and the global best achieved by the entire swarm. The optimization procedure consists of spreading the information about good solutions through the swarm so that particles move over the space under a

velocity vector biased by the positions of the aforementioned best solutions in the swarm.

- GA [19] are heuristic solvers inspired from observed processes in the genetic inheritance among generations of individuals. The main stages of the algorithm are selection, crossover, mutation and evaluation. The search technique begins with a randomly generated initial population of individuals, from which a number of breeding solutions or parents are selected based on their fitness values. Then, in the crossover stage two individuals are taken randomly from the selected population and combined to yield offspring solutions. Finally, each offspring solution undergoes small perturbations of its compounding variables under probability p_m. All offspring solutions are then evaluated, replacing the previous population. The procedure is repeated until the termination criteria is met.
- EDAs [20] are population-based optimization algorithms that guide the search by sampling promising solutions from learned generative probabilistic models, i.e. In EDAs new individuals are sampled from a probability distribution estimated from the previous generation of solutions and their associated fitness values. In this work we assume for simplicity a canonical EDAs where optimization variables are assumed to be independent from each other in the probabilistic model.

III. Experimental Results and Discussion

Our approach will be validated over different datasets utilized in the literature related to time series classification. In particular we will use the articulatory word [21], cricket and Auslan (sign language used by the Australian deaf community) [23] datasets. It is important to remark that for all datasets time series have been normalized via Z-score so that every dimension has zero mean and unit standard deviations, i.e., $\forall d \in \{1,\ldots,D\}$ of any given time series $\mathbf{T}$ of every dataset,

$$\mathbf{t}^{(d)}_{\text{norm}} = \frac{\mathbf{t}^{(d)} - \text{E}[\mathbf{t}^{(d)}]}{\sqrt{\text{Var}[\mathbf{t}^{(d)}]}}, \tag{10}$$

where $\text{E}[\cdot]$ and $\text{Var}[\cdot]$ denotes expectation and variance, respectively. In the following list a brief description of each data set and the performed experiments is provided:

- The Auslan dataset comprises 95 different signs performed by 5 signers, yielding a total of 6648 time series, each with $d = 15$ dimensions [24]. From these dimensions x, y, z and $roll$ attributes have been selected as predictors for a small subset of the overall dataset corresponding to sequences labeled with *all*, *answer*, *boy*, *buy*, *cold*, *come*, *crazy*, *different*, *exit* and *forget*. On the one hand, x, y and z variables record the up-down, right/left and forward/backward movements of the signers' hands, respectively. However, they should not be taken to form orthogonal basis, hence relations among variables are expected. On the other hand, the $roll$ dimension tracks the palm rotation.
- The cricket dataset consist of a collection of 12 referee signals, each with ten repetitions. The data contains observations of x, y and z axes motion measured with an accelerometer placed on both, left and right wrists of the umpires. As in [10] we will use different dimension pairs to predict each signal.
- The articulatory word dataset contains tongue, lips and head motion (using 12 sensors) of native English speakers performing 25 different words. Altogether the dataset amounts up to 575 time series, each comprising x, y and z position of each sensor. From the total of $D = 36$ available dimensions, we will use different combinations considering the sensors on the tongue tip (T1), the upper (UL) and the lower lip (LL) as done in [10].

Regarding the nature-inspired solvers a summary of the specific parameter values utilized for each method are listed below. It must be noted that the same termination criterion is utilized for all algorithms in the benchmark, i.e. the algorithm is forced to stop when the fitness value of the best proposed solution does not improve for max_{gen} generations.

- In the i-th iteration of the SA solver, the acceptance of a new solution $\boldsymbol{\omega}^{i+1} = \boldsymbol{\omega}^i + n$ – where n is a random variable given by a standard normal distribution, $\mathcal{N}(0,1)$ – is ruled by $\boldsymbol{\omega}^{i+1}$, $\boldsymbol{\omega}^i$ and a temperature parameter T_i that jointly define the solution acceptance probability. The temperature of the algorithm is enforced to go from value 1 to 0 along the iterations of the algorithm as $T_{i+1} = \xi T_i$ where $0 \leq \xi < 1$ is the cooling rate. In all simulations ξ and max_{gen} have been set to 0.1 and 100, respectively.
- In PSO the parameters of the algorithm have been chosen so that the movement of each particle in the search space $\boldsymbol{\omega}$ is governed by $\boldsymbol{\omega}_{i+1} = \boldsymbol{\omega}_i + \boldsymbol{\alpha}_{\text{indv}} + \boldsymbol{\alpha}_{\text{global}} + \boldsymbol{\alpha}_{\text{neigh}}$, where $i \in \{1,\ldots,I\}$ denotes the iteration index and the vectors in the right side $\boldsymbol{\alpha}_{\text{indv}} = 0.5(\boldsymbol{\omega}_i - \boldsymbol{\omega}_{i-1})$, $\boldsymbol{\alpha}_{\text{global}} = 2.1 n_{\text{global}}(\boldsymbol{\omega}_{\text{global}} - \boldsymbol{\omega}_i)$ and $\boldsymbol{\alpha}_{\text{neigh}} = 2.1 n_{\text{neigh}}(\boldsymbol{\omega}_{\text{neigh}} - \boldsymbol{\omega}_i)$ correspond to the tendency to move towards the previous position, the influence to move towards the entire swarm best solution, $\boldsymbol{\omega}_{\text{global}}$, and the influence to move towards the neighbours best solution, $\boldsymbol{\omega}_{\text{neigh}}$, respectively. The maximum number of generations, max_{gen} is set to 100 while n_{global} and n_{best} are realizations of a continuous random variable uniformly distributed in the range $[0, 1]$.
- In GA the population size is 40 individuals, from which the number of solutions in the selection step have been set to 20. In the crossover stage, the chosen operator is single-point crossover with probability $p_c = 0.9$. Finally the mutation of each offspring element $\omega_d \in \boldsymbol{\omega}$ is changed to $\omega_d + n$ with mutation probability $p_m = 0.1$ where $n \sim \mathcal{N}(0,1)$. As in previous solvers the stopping criteria have been set to $\text{max}_{\text{gen}} = 100$.
- In EDA the offspring values $\boldsymbol{\omega}^i$ of the i-th generation are drawn from a multidimensional Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}^i, \boldsymbol{\Sigma}^i)$ whose mean vector $\boldsymbol{\mu}^i$ and covariance matrix $\boldsymbol{\Sigma}^i$ are given by

$$\boldsymbol{\mu}^i = (\mu^i_1,\ldots,\mu^i_d,\ldots,\mu^i_D),\ \mu^i_d = \text{E}[\boldsymbol{\omega}^{i-1}_d] \tag{11}$$

$$\boldsymbol{\Sigma}^i = \text{diag}(\Sigma^i_1,\ldots,\Sigma^i_d,\ldots,\Sigma^i_D)\ :\ \Sigma_d = \text{Var}[\boldsymbol{\omega}^{i-1}_d] \tag{12}$$

TABLE I: Average, first quartile and third quartile scores for the simulated datasets and heuristic wrappers.

Dataset label	Variables	DTW_I	DTW_D	$\text{DTW}_{opt}^{\text{SA}}$	$\text{DTW}_{opt}^{\text{PSO}}$	$\text{DTW}_{opt}^{\text{GA}}$	$\text{DTW}_{opt}^{\text{EDAs}}$
AUSLAN-XYZR	$x, y, z, roll$	0.757, (0.714 − 0.786)	0.789, (0.750 − 0.812)	0.800, (0.786 − 0.848)	**0.814**, (0.768 − 0.848)	0.796, (0.786 − 0.812)	0.793, (0.759 − 0.821)
AUSLAN-XYZ	x, y, z	0.805, (0.726 − 0.857)	0.614, (0.583 − 0.667)	0.805, (0.726 − 0.857)	0.800, (0.726 − 0.857)	**0.814** (0.768 − 0.848)	0.776, (0.714 − 0.845)
CRICKET-XRXL	x_{right}, x_{left}	0.946, (0.917 − 0.958)	0.921, (0.917 − 0.927)	**0.971**, (0.958 − 1.000)	**0.971**, (0.958 − 1.000)	0.971, (0.958 − 1.000)	**0.971**, (0.958 − 1.000)
CRICKET-XRYL	x_{right}, y_{left}	0.954, (0.917 − 0.990)	0.954, (0.917 − 1.000)	**0.975**, (0.958 − 1.000)	0.971, (0.958 − 1.000)	0.958, (0.927 − 0.990)	0.967, (0.958 − 1.000)
CRICKET-YRXL	y_{right}, x_{left}	0.988, 1.000, −1.000	0.983, (1.000 − 1.000)	**0.992**, (1.000 − 1.000)	**0.992**, (1.000 − 1.000)	**0.992**, (1.000 − 1.000)	0.983, (1.000 − 1.000)
CRICKET-YRYL	y_{right}, y_{left}	0.969, (0.979 − 1.000)	0.988, (1.000 − 1.000)	**0.992**, (1.000 − 1.000)	0.988, (1.000 − 1.000)	**0.992**, (1.000 − 1.000)	**0.992**, (1.000 − 1.000)
CRICKET-ZRZL	z_{right}, z_{left}	0.954, (0.917 − 1.000)	0.975, (0.938 − 1.000)	**0.979**, (0.969 − 1.000)	**0.979**, (0.969 − 1.000)	0.971, (0.969 − 1.000)	0.971, (0.969 − 1.000)
ARTI-UXTZ	UL_x, $T1_z$	0.839, (0.830 − 0.860)	0.863, (0.845 − 0.880)	0.891, (0.883 − 0.907)	**0.902**, (0.900 − 0.920)	0.894, (0.880 − 0.920)	0.894, (0.880 − 0.917)
ARTI-TZLZ	$T1_z$, LL_z	0.895, (0.880 − 0.917)	0.941, (0.923 − 0.960)	0.938, (0.923 − 0.957)	0.940, (0.935 − 0.945)	0.942, (0.935 − 0.960)	**0.944**, (0.923 − 0.960)
ARTI-TZLY	$T1_z$, LL_y	0.887, (0.880 − 0.917)	**0.933**, (0.920 − 0.955)	**0.933**, (0.923 − 0.955)	0.923, (0.900 − 0.938)	0.926, (0.920 − 0.940)	0.926, (0.920 − 0.938)
ARTI-LXTYTZ	LL_x, $T1_y$, $T1_z$	0.923, (0.885 − 0.960)	0.954, (0.950 − 0.978)	**0.966**, (0.945 − 0.985)	0.965, (0.945 − 0.978)	0.963, (0.942 − 0.978)	0.965, (0.945 − 0.987)
ARTI-TYTZ	$T1_y$, $T1_z$	0.892, (0.883 − 0.917)	**0.949**, (0.925 − 0.960)	0.944, (0.923 − 0.960)	0.945, (0.925 − 0.960)	0.943, (0.925 − 0.960)	0.945, (0.925 − 0.960)
ARTI-TYLZ	$T1_y$, LL_y	0.863, (0.840 − 0.880)	0.919, (0.902 − 0.940)	**0.924**, (0.902 − 0.940)	0.921, (0.900 − 0.940)	**0.924**, (0.900 − 0.955)	**0.924**, (0.900 − 0.955)
ARTI-TXTYUZ	$T1_x$, $T1_y$, UL_y	0.912, (0.900 − 0.920)	0.931, (0.920 − 0.945)	0.944, (0.940 − 0.957)	0.939, (0.925 − 0.957)	0.940, (0.940 − 0.947)	**0.947**, (0.940 − 0.960)
ARTI-TXTYTZ	$T1_x$, $T1_y$, $T1_z$	0.928, (0.897 − 0.955)	0.940, (0.920 − 0.960)	**0.945**, (0.920 − 0.960)	0.942, (0.920 − 0.972)	**0.945**, (0.923 − 0.975)	0.939, (0.925 − 0.957)

where $\boldsymbol{\omega}_d^{i-1} = \{\omega_d^{1,i-1}, \ldots, \omega_d^{100,i-1}\}$ corresponds to the vector collecting the values of the d-th variable along a 100-sized population of individuals drawn from $\mathcal{N}(\boldsymbol{\mu}^{i-1}, \boldsymbol{\Sigma}^{i-1})$ at generation $i-1$.

A. Discussion on Predictive Score Results

Besides the computation of a performance estimate to evaluate the fitness of the $\boldsymbol{\omega}$ parameters proposed by the heuristic wrapper, the goodness of the optimized classifier when trained over $\text{DTW}_{opt}(\cdot,\cdot)$ should be measured over unseen test data. For this reason we will use again stratified K-fold cross validation to first split the entire dataset in training and test sets. Algorithm 2 drafts, for K partitions, the procedure to compute the goodness measure. In essence, training and test sets are constructed using $K-1$ folds for the former and the left-out fold for the latter. For all possible training-test combinations, optimized $\boldsymbol{\omega}$ values are found using an nature-inspired algorithms (line 6). The average of the K computed scores for the train-test splits will yield a measure of the expected performance of the proposed wrapper when facing new test samples.

Algorithm 2 Computation of the test score

1: **procedure** TESTSCORE(K, dataset)
2: Split dataset into K stratified folds
3: **for** $i \in \{1, \ldots, K\}$ **do**
4: Set the i-th fold as the test set
5: Set the remaining folds as the training set
6: Optimize $\boldsymbol{\omega}$ using the training set, a heuristic solver and the fitness in Algorithm 1
7: Predict test set labels with the classifier using $\text{DTW}_{opt}(\cdot,\cdot)$ with the optimized $\boldsymbol{\omega}$
8: Compute performance score of this fold as η_i
9: **end for**
10: **return** K-cv test score as $\sum_{i=1}^{K} \eta_i / K$
11: **end procedure**

Table I shows the estimated accuracy rates as well as first and third quartile rates obtained by the independent ($\text{DTW}_I(\cdot,\cdot)$), dependent ($\text{DTW}_D(\cdot,\cdot)$) and optimized ($\text{DTW}_{opt}(\cdot,\cdot)$) models for each solver. The bolded value in the table indicates the best average accuracy rate among all designed models. From these scores several observations can be pointed out. To begin with, all wrappers have similar performance being SA slightly more accurate than the rest of the heuristic methods. Results for the Auslan dataset suggest that $\text{DTW}_{opt}(\cdot,\cdot)$ is more resilient to the selection of the dataset variables, specially with SA and PSO algorithms, than $\text{DTW}_I(\cdot,\cdot)$ and $\text{DTW}_D(\cdot,\cdot)$. Cricket scores are complex to analyse because all distance models achieve high accuracies over the simulated variable combinations. An exception is the CRICKET-XRXL case, for which the predictive score with the optimized distance is higher than its dependent and independent counterparts for all utilized wrappers. Regarding the rest of datasets, lower performance gains are noted; we can conclude that in general, our model is at least as accurate as the best among $\text{DTW}_I(\cdot,\cdot)$ and $\text{DTW}_D(\cdot,\cdot)$ regardless the method we use.

B. Discussion on Optimized Coefficients Values

Although average accuracies allow comparing our model with $\text{DTW}_I(\cdot,\cdot)$ and $\text{DTW}_D(\cdot,\cdot)$ in terms of model fitness, they do not provide any insight on the statistical distribution of such scores, nor do they shed any light on the values of their associated weights $\boldsymbol{\omega}$. A further analysis of the distribution of 10-fold cross-validation accuracy scores and the optimized values of $\{\omega_1, \ldots, \omega_D\}$ values is done in Figures 2.a through 2.f in the form of boxplots. The subset of simulated cases represented in these figures is a representative sample, that illustrate best, the casuistry that occurs in all performed experiments.

Figures 2.a and 2.d illustrate the outcomes of the PSO solver where $\text{DTW}_{opt}(\cdot,\cdot)$ outperforms both $\text{DTW}_D(\cdot,\cdot)$ and $\text{DTW}_I(\cdot,\cdot)$, with non-zero values for all $\{\omega_1, \ldots, \omega_D\}$. This indicates that the relations between dimensions are important for classification. Moreover, to verify that the performance gaps for $\text{DTW}_{opt}(\cdot,\cdot)$ and those of $\text{DTW}_D(\cdot,\cdot)$ and $\text{DTW}_I(\cdot,\cdot)$ results are statistically significant we have performed a non-parametric Wilcoxon signed-rank test to check whether result samples come from distribution with different medians. Since the obtained p-value falls below 0.05 for both cases the hypothesis of statistical significance is confirmed. In particular, we get p-value $= 0.007$ when the test is performed with $\text{DTW}_{opt}(\cdot,\cdot)$ and $\text{DTW}_D(\cdot,\cdot)$ score samples and p-value $= 0.01$ with $\text{DTW}_{opt}(\cdot,\cdot)$ and $\text{DTW}_I(\cdot,\cdot)$. Although we have mentioned before that SA is slightly more accurate, note that for this particular case, PSO is among all solvers the procedure with highest accuracy rates.

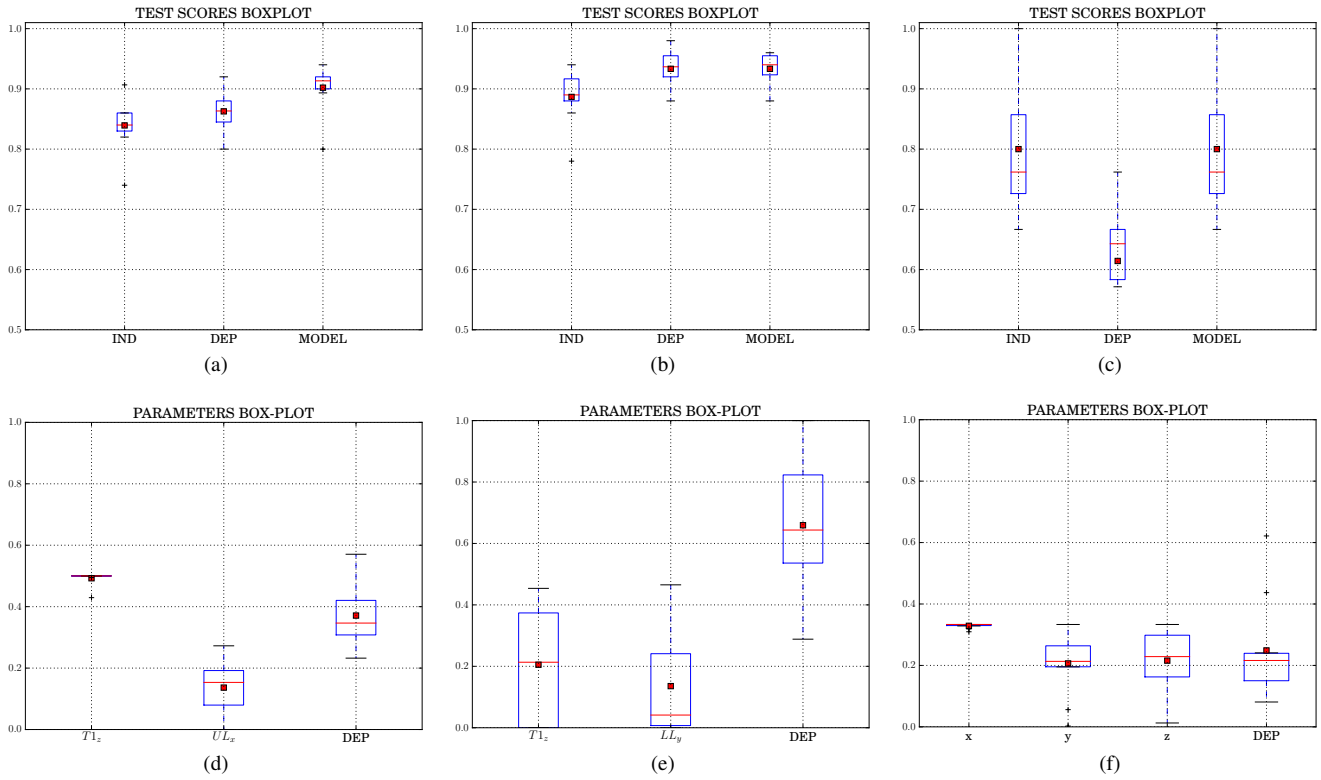

Fig. 2: Boxplot corresponding to the test scores obtained via stratified 10-fold corresponding to the ARTI−TUXTZ (a), ARTI−TZLY (b) and AUSLAN−XYZ (c) datasets, along with boxplots showing the distribution of the obtained weights $\boldsymbol{\omega}$ for each case – (d), (e) and (f), respectively. The red square indicates the mean value of the sample.

The discussion follows by Figures 2.b and 2.e, which exemplifies, with the ARTI−TZLY dataset and SA solver, the case when $\text{DTW}_{opt}(\cdot,\cdot)$ and $\text{DTW}_D(\cdot,\cdot)$ render a similar predictive performance. As could be a priori expected, the optimized (ω_1, ω_2) weights that gauge the contribution of each dimension in isolation to the optimized distance in (9) are close to zero in contrast to the dependet part contribution. By contrast, SA results for the AUSLAN−XYZ dataset (Figures 2.c and 2.f) unveil an identical performance of the independent and optimized distance model and a notably worse behavior of $\text{DTW}_D(\cdot,\cdot)$. One would accordingly expect high values for the weights $\{\omega_1, \omega_2, \omega_3\}$ (close to $1/D = 1/3$) so that the independent part in (9) dominates over $\text{DTW}_D(\cdot,\cdot)$. This does not hold in the plotted results, where the the contribution of both independent and dependent parts are similar; even more, the dependent contribution is never negligible. The rationale behind this contradictory effect might lie on the tight coupling among variables imposed in the search for the warping path in $\text{DTW}_D(\cdot,\cdot)$. As imposed by the proposed definition of $\text{DTW}_{opt}(\cdot,\cdot)$ the optimization of the contribution of the dependent part to this combined metric does not discriminate between dimensions. Taking into account variability of y and and more specifically z dimensions weights, Figure 2.f, indicates that $\text{DTW}_D(\cdot,\cdot)$ part is somehow compensated in order to reduce the lack of predictability of one of both variables equally weighted inside the dependent part. This observation suggests that a generalization of the inner pairwise distance between samples $\mathbf{t}_i$ and $\mathbf{u}_j$ to allow for the optimization of each variable to the dependent DTW distance should overcome this issue and achieve better accuracies.

IV. Conclusions and Future Research Lines

In this paper we have defined a simple similarity measure for multidimensional time series classification that leverages the ability to accommodate time warps featured by the DTW distance and takes into account the existing relations among dimensions. The proposed scheme is based on a heuristic wrapper that optimizes the values of the weights balancing the contribution of independent and dependent DTW distances to the proposed measure. The optimization criterion is based on the maximization of the cross-validated prediction score of a distance-based classifier operating on the similarities iteratively refined by the heuristic. A benchmark of four heuristic solvers have been utilized SA, PSO, GA and EDAs. When assessed over several datasets from the literature with the mentioned heuristic solvers we have shown on one hand that our proposed distance model with a 1-NN classifier performs, in general, equal or better than the same learner with independent and dependent DTW distances. On the other hand, we have seen that there are not much difference in heuristic

solver performance although in our case SA seems to return slightly more accurate results.

Although the defined distance has proven to be competitive, there are some open research paths that should be addressed to further improve its predictive performance. We have seen that the adaptability of the independent part is higher than that of the dependent part. To ensure the variability of the dependent part in equation (9), the definition of the inner distance in $\mathrm{DTW}_D(\cdot,\cdot)$ should allow weighting differently each dimension in the warping path discovery process as in [11] trying to decrease additionally, the computation cost as they conclude in their work. Other error rate functions such as like larger-margin nearest neighbor formulation, which is often utilized in distance learning tasks, [25] will also be studied.

ACKNOWLEDGMENTS

This work has been supported by the Basque Government through the ELKARTEK program (ref. BID3ABI). Aritz Pérez is partially supported by the ELKARTEK program from the Basque Government, and by the Spanish Ministry of Economy and Competitiveness MINECO: BCAM Severo Ochoa excellence accreditation SVP-2014-068574 and SEV-2013-0323.

REFERENCES

[1] D. C. Montgomery, C. L. Jennings, M. Kulahci, "Introduction to Time Series Analysis and Forecasting," John Wiley & Sons, 2015.

[2] A. Kazem, E. Sharifi, F. K. Hussain, M. Saberi, O. K. Hussain, "Support vector regression with chaos-based firefly algorithm for stock market price forecasting," Applied soft computing, Vol.13, N. 2, pp. 947-958, 2013.

[3] A. Kampouraki, G. Manis, C. Nikou, "Heartbeat time series classification with support vector machines," IEEE Transactions on Information Technology in Biomedicine, Vol. 13, N. 4, pp. 512-518, 2009.

[4] W. A. Chaovalitwongse, Y. J. Fan, R. C. Sachdeo, "On the time series k-nearest neighbor classification of abnormal brain activity," IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, Vol. 37, N. 6, pp. 1005-1016, 2007.

[5] F., Lotte, M. Congedo, A. Lécuyer, F. Lamarche, B. Arnaldi, "A review of classification algorithms for EEG-based braincomputer interfaces," Journal of neural engineering, Vol. 4, N. 2, pp. R1, 2007.

[6] Z. Xing, J. Pei, E. Keogh, "A brief survey on sequence classification," ACM Sigkdd Explorations Newsletter, Vol. 12, N. 1, pp. 40-48, 2010.

[7] T. W. Liao, "Clustering of Time Series Data – a Survey," Pattern Recognition, Vol. 38, N. 11, pp. 1857-1874, 2005.

[8] T. Rakthanmanon, B. Campana, A. Mueen, G. Batista, B. Westover, Q. Zhu, J. Zakaria, E. Keogh, "Searching and mining trillions of time series subsequences under dynamic time warping," In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp. 262-270, 2012.

[9] Z. Bankó, J. Abonyi, "Correlation Based Dynamic Time Warping of Multivariate Time Series," Expert Systems with Applications, Vol. 39, N. 17, pp. 12814-12823, 2012.

[10] M. Shokoohi-Yekta, B. Hu, H. Jin, J. Wang, E. Keogh, "Generalizing Dynamic Time Warping to the Multi-Dimensional Case Requires an Adaptive Approach," Data Mining and Knowledge Discovery, Vol. 31, N. 1, pp. 1-31, 2017.

[11] J. Mei, M. Liu, Y. F. Wang, H. Gao, "Learning a mahalanobis distance-based dynamic time warping measure for multivariate time series classification," IEEE transactions on Cybernetics, Vol. 46, N. 6, pp. 1363-1374, 2016.

[12] T. Górecki, M. Łuczak, "Multivariate Time Series Classification with Parametric Derivative Dynamic Time Warping," Expert Systems with Applications, Vol. 42, N. 5, pp. 2305-2312, 2015.

[13] M. R. Peterson, T. E. Doom, M. L. Raymer, "Ga-facilitated knn classifier optimization with varying similarity measures" In Evolutionary Computation, 2005, The 2005 IEEE Congress on, Vol. 3, pp. 2514-2521, IEEE.

[14] H. Sakoe, S. Chiba, "Dynamic Programming Algorithm Optimization for Spoken Word Recognition," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 26, N. 1, pp. 43-49, 1978.

[15] F. Itakura, "Minimum prediction residual principle applied to speech recognition," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 23, N. 1, pp. 67-72, 1975.

[16] J. D. Rodriguez, A. Pérez, J. A. Lozano, "A General Framework for the Statistical Analysis of the Sources of Variance for Classification Error Estimators," Pattern Recognition, Vol. 46, N. 3, pp. 855-864, 2013.

[17] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi, "Optimization by Simulated Annealing," Science, Vol. 220, N. 4598, pp. 671-680, 1983.

[18] J. Kennedy, " Particle swarm optimization," In Encyclopedia of machine learning, pp. 760-766. Springer US, 2011.

[19] D. Whitley, "A genetic algorithm tutorial," Statistics and computing, Vol. 4, N. 2, pp. 65-85, 1994.

[20] P. Larrañaga, J. A. Lozano, eds., "Estimation of distribution algorithms: A new tool for evolutionary computation," Vol. 2, Springer Science & Business Media, 2001.

[21] J. Wang, A. Balasubramanian, L. Mojica de La Vega, J. R. Green, A. Samal, B. Prabhakaran, "Word Recognition from Continuous Articulatory Movement Time-Series Data using Symbolic Representations," ACL/ISCA Interspeech Workshop on Speech and Language Processing for Assistive Technologies, pp. 119-127, 2013.

[22] Y. Chen, E. Keogh, B. Hu, N. Begum, A. Bagnall, A. Mueen, G. Batista, "The UCR Time Series Classification Archive," www.cs.ucr.edu/~eamonn/time_series_data/, 2015.

[23] M. Lichman, "UCI Machine Learning Repository," http://archive.ics.uci.edu/ml, 2013.

[24] M. W. Kadous, "Temporal Classification: Extending the Classification Paradigm to Multivariate Time Series," Doctoral Dissertation, University of New South Wales, 2002.

[25] K. Q. Weinberger, L. K. Saul, "Distance Metric Learning for Large Margin Nearest Neighbour Classification," Journal of Machine Learning Research, Vol. 10, pp. 207-244, 2009.