Pattern Recognition 43 (2010) 3787–3794

Contents lists available at ScienceDirect

Pattern Recognition

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

Combining discrete SVM and fixed cardinality warping distances for multivariate time series classification

C. Orsenigo *, C. Vercellis

Dipartimento di Ingegneria Gestionale, Politecnico di Milano, Via Lambruschini 4b, 20156 Milano, Italy

ARTICLE INFO

Article history:
Received 27 March 2009
Received in revised form
23 February 2010
Accepted 7 June 2010

Keywords:
Time series classification
Support vector machines
Discrete support vector machines
Learning theory
Warping distance

ABSTRACT

Time series classification is a supervised learning problem aimed at labeling temporally structured multivariate sequences of variable length. The most common approach reduces time series classification to a static problem by suitably transforming the set of multivariate input sequences into a rectangular table composed by a fixed number of columns. Then, one of the alternative efficient methods for classification is applied for predicting the class of new temporal sequences. In this paper, we propose a new classification method, based on a temporal extension of discrete support vector machines, that benefits from the notions of warping distance and softened variable margin. Furthermore, in order to transform a temporal dataset into a rectangular shape, we also develop a new method based on fixed cardinality warping distances. Computational tests performed on both benchmark and real marketing temporal datasets indicate the effectiveness of the proposed method in comparison to other techniques.

1. Introduction

Time series classification is a supervised learning problem aimed at labeling temporally structured multivariate sequences of variable length. Several applications have been naturally cast in the form of time series classification, such as labeling the trajectories of vehicles monitored by video surveillance systems, or indexing ECG diagrams in a medical diagnosis context. However, there are also application domains where the temporal nature of the data is less evident and has been usually neglected. In particular, a large majority of classification problems arising in the domain of relational marketing are based on sequential data: the behavior of customers is observed through time, and their interactions with the company actually represent multivariate time series. For example, for a telecommunication company each time series corresponds to the transactions of a single customer recorded along time periods, described by multiple variables which may represent duration, economic value and number of calls made for different types of connections. To model and predict customers loyalty, one can formulate a binary classification problem in which the label of each time series indicates whether customer is still active or is a *churner* who presumably left the company in favour of a competitor. When dealing with marketing data, it is a common practice to reduce them to tabular shapes by simply consolidating the variables on vertical time frames along the temporal dimension. Other classification problems involve instead univariate time series. This is the case of ECG diagrams labeling, where each temporal sequence expresses the cardiac activity recorded by one electrode and is classified according to the patient state of health. We believe that properly framing classification problems within a temporal setting may lead to higher accuracy in prediction, as shown by our computational experiences.

Several alternative paradigms for time series classification have been proposed in the literature; we refer the reader to [1] for a review. The most common approach is based on a two-stage procedure. First, a rectangular representation of the time series is derived by suitably transforming the set of multivariate input sequences into a fixed number of columns, through different rectangularization mechanisms. Then, a classification method is applied for labeling the data, such as support vector machines (SVM) [2], neural networks [3], induction trees [4], among others. The rectangularization process is aimed at compressing the temporal span of the different time series without compromising the wealth of information contained in the original data. Consequently, it plays a significant role in the whole classification task and should not be undervalued.

An entirely different approach, which is very effective for univariate time series classification, is based on the notion of warping distance, a suitable measure of similarity between pairs of time series. This distance allows to detect clusters and to predict with high accuracy the class of new temporal sequences by using distance-based methods, such as the 1-nearest neighbor

* Corresponding author. Tel.: +39 02 2399 3970; fax: +39 02 2399 3978.
E-mail addresses: carlotta.orsenigo@polimi.it (C. Orsenigo), carlo.vercellis@polimi.it (C. Vercellis).

0031-3203/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2010.06.005

classifier [5,6]. Kernels based on dynamic time warping and incorporated within traditional SVMs have been proposed in [7–9] and recently applied in [10] to brain activity classification.

In this paper, we propose a new classification method framed within the two-stage scheme outlined above, that benefits from the powerful notion of warping distance in both phases. More specifically, we develop a new temporal variant of discrete support vector machines to perform the classification task. Discrete SVM, first introduced in [11,12], are a successful alternative to classical SVM based on the idea of accurately evaluating the number of misclassified examples instead of measuring their distance from the separating hyperplane. Starting from the original formulation, discrete SVM have been successfully extended in several directions, to deal with multi-class problems [13] or to learn from a small number of training examples [14].

The optimization model representing temporal discrete SVM has two main novelties. First, it explicitly takes into account the overall similarity among time series assigned to the same class, by including into the objective function a term that depends on the warping distances. Furthermore, the optimal discriminating hyperplane derived by the model establishes a variable softening of the margin of separation by introducing an additional term into the objective function and modifying some of the constraints of the optimization problem. The explicit inclusion of the margin as a variable allows to regulate more effectively the trade-off between the misclassification error on the training data and the generalization capability, by means of the corresponding cost coefficient.

A new procedure is also proposed for the rectangularization stage, in order to fully exploit the intrinsic temporal dependence in the data, by considering a fixed cardinality variant of the warping distance that can be efficiently computed. By this way, the dependence from time is preserved in the derived tabular shape, and a proper phasing and alignment of the time series is achieved. For example, in marketing applications customers with different lifetime profiles are aligned and phased, allowing to extract the maximum amount of information carried through their recorded behavior.

To evaluate the effectiveness of the proposed method, which combines the rectangularization phase with the classification task performed by temporal discrete SVM, we have considered six datasets, mostly composed by multivariate temporal sequences. The results seem to indicate that temporal discrete SVM have a great potential to perform an accurate classification of multivariate time series.

2. Temporal classification and warping distance

In a classification problem, that will be termed *static* in the sequel to underline the difference from *temporal* classification problems defined below, a set $\mathcal{S}_m=\{(\mathbf{x}_i,y_i), i\in\mathcal{M}=\{1,2,\ldots,m\}\}$ of training input–output *examples* is given. Here $\mathbf{x}_i\in\Re^n$ is an input vector of real numbers and $y_i\in\mathcal{D}=\{1,2,\ldots,D\}$ is the categorical *class label* associated to $\mathbf{x}_i$. Each component x_{ij} of an example $\mathbf{x}_i$ is thought to be a realization of a random variable B_j, $j\in\mathcal{N}=\{1,2,\ldots,n\}$ that will be referred to as an *attribute* of $\mathcal{S}_m$. Let $\mathcal{H}$ denote a set of functions $f:\Re^n\mapsto\mathcal{D}$ that represent hypothetical relationships between $\mathbf{x}_i$ and y_i. A *static classification problem* consists of defining an appropriate hypotheses space $\mathcal{H}$ and a function $f^*\in\mathcal{H}$ which optimally describes the relationship between inputs and outputs. When there are only two classes, i.e. $D=2$, we obtain a *binary* classification problem, while the general case is termed as *multicategory* classification. For binary problems we assume that $y_i\in\{-1,1\}$, without loss of generality.

In a temporal classification problem we are given a set of multivariate time series $\{\mathbf{A}_i\}$, $i\in\mathcal{M}$, where each $\mathbf{A}_i=[a_{ilt}]$ is a rectangular matrix of size $L\times T_i$. Here $l\in\mathcal{L}=\{1,2,\ldots,L\}$ is the index associated to the attributes of the time series, whereas $t\in\mathcal{T}_i=\{1,2,\ldots,T_i\}$ is the temporal index that may vary in a different range for each $\mathbf{A}_i$. Every time series is also associated with a class label $y_i\in\mathcal{D}$. The *temporal classification problem* consists of defining an appropriate function f^* which optimally describes the relationship between the time series $\{\mathbf{A}_i\}$ and their labels $\{y_i\}$, in the sense of minimizing some measure of misclassification.

Hence, the main difference between static and temporal classification problems lies in the native rectangular structure of the former, opposed to the variable length of each record in the latter. Due to the vast amount of alternative effective methods available for static classification problems, a commonly adopted approach to time series classification relies on a two-phase procedure. First, an appropriate transformation is devised to obtain a rectangular representation of the set $\{\mathbf{A}_i\}$. Then, a method for static classification is applied to the rectangular dataset derived in the first phase. An entirely different approach to time series classification is based on the notion of warping distance described in the next subsection.

2.1. Warping distance

The *warping distance*, originally introduced in the context of speech recognition and signal processing [15], has been successfully applied as a proximity measure for clustering and labeling univariate time series [5,6,16]. As a similarity measure, the warping distance has proven to be more robust and versatile than the Euclidean metric since, unlike this latter, it copes with sequences of variable length and automatically performs shifts in the sequences to identify similar profiles with different phases. Furthermore, it has been shown that the warping distance for each pair of time series can be calculated efficiently by dynamic optimization in $O(T_{max}^2)$ time, where $T_{max}=\max\{T_i : i\in\mathcal{M}\}$ is the maximum temporal length of the m time series. Let also $T_{min}=\min\{T_i : i\in\mathcal{M}\}$ be the minimum length.

We start by defining the warping distance for univariate time series, for which $L=1$, where a single attribute is recorded for each sequence along its time trajectory. Notice that our description departs from the way the argument is usually developed in the literature, where the warping distance is introduced in strict connection to the dynamic programming procedure traditionally adopted for its computation. Instead, we prefer to express the evaluation of the warping distance as an optimal path problem, since this formulation sheds a clear light on the problem structure, and allows to derive more easily the fixed cardinality extension proposed in Section 3.

In order to find the optimal alignment between two univariate time series $\mathbf{A}_i$ and $\mathbf{A}_k$, let $G=(V,E)$ be a directed graph whose vertices in V correspond to the pair of time periods (r,s), $r\in\mathcal{T}_i$, $s\in\mathcal{T}_k$. A vertex $v=(r,s)$ indicates that the r-th value of the time series $\mathbf{A}_i$ is matched with the s-th value of $\mathbf{A}_k$. An oriented arc (u,v) connects vertex $u=(p,q)$ to vertex $v=(r,s)$ if and only if one of the following mutually exclusive conditions holds:

$$\{r=p+1,s=q\}\vee\{r=p+1,s=q+1\}\vee\{r=p,s=q+1\}. \tag{1}$$

Consequently, each vertex $u\in G$ has at most three outgoing arcs, associated to the three conditions described in (1) and illustrated in Fig. 1.

The length γ_{uv} of the arc (u,v), connecting the vertices $u=(p,q)$ and $v=(r,s)$, is defined as the squared distance associated to the potential alignment of period r in $\mathbf{A}_i$ to period s in $\mathbf{A}_k$, given by

$$\gamma_{uv}=\overline{\gamma}_{ik}(r,s)=(a_{i1r}-a_{k1s})^2. \tag{2}$$

Let also $v_f=(1,1)$ and $v_l=(T_i,T_k)$ be the vertices corresponding to the alignment of the first and last periods in the two sequences, respectively.

A *warping path* in G is any path connecting the source vertex v_f to the destination vertex v_l. It is clear that a warping path determines a coherent alignment between the two sequences of periods composing each time series, such that matched time periods are monotonically spaced in time and contiguous. The *warping distance* between the time series $\mathbf{A}_i$ and $\mathbf{A}_k$ is defined as the length of the shortest warping path in G (Fig. 2). The cardinality H of the shortest warping path lies in the interval $[\max(T_i,T_k),T_i+T_k-1]$. Intuitively, the warping distance is a suitable measure of similarity that allows to achieve a proper phasing and alignment of two time series, as shown in Fig. 3.

Turning to multivariate time series, the concept of warping path and warping distance between $\mathbf{A}_i$ and $\mathbf{A}_k$ can be readily generalized by summing over the L attributes, defining the length γ_{uv} of the arc (u,v) as

$$\gamma_{uv} = \sum_{l=1}^{L} (a_{ilr} - a_{kls})^2. \tag{3}$$

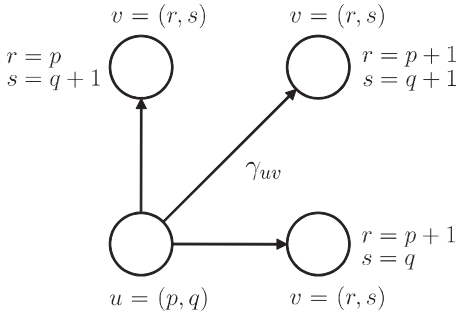

Fig. 1. Possible connections between the vertices $u=(p,q)$ and $v=(r,s)$ in G.

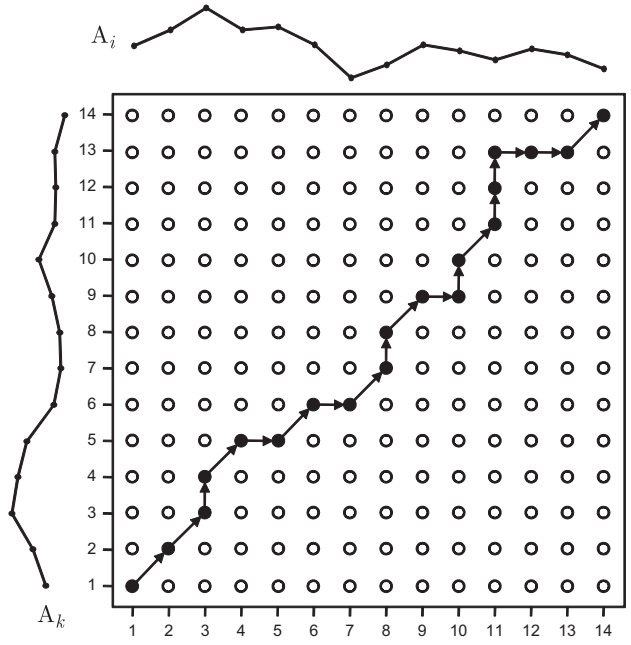

Fig. 2. Shortest warping path between two univariate time series, $\mathbf{A}_i$ and $\mathbf{A}_k$. In the example, both series have the same length: $T_i=T_k=14$.

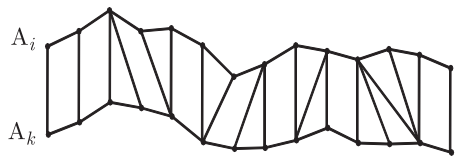

Fig. 3. Alignment between $\mathbf{A}_i$ and $\mathbf{A}_k$.

2.2. A DAGSP algorithm for warping distance computation

The graph G is a weighted directed acyclic graph (DAG) and is also rather sparse since the outdegree of its vertices is at most three, so that $|E|\le 3|V|$. It is known [17] that for a DAG the length z_v of the single-source shortest path between v_f and all other vertices $v\in V$ can be computed in time $O(|E|+|V|)$, reducing to $O(|V|)$ for G, by the procedure DAGSP:

procedure DAGSP (G,v_f)
1 topologically sort the vertices of G
2 **do** $z_{v_f}=0$; **for** each $u\in V$ **do** $z_u=\infty$
3 **for** each $u\in V$ in topological order
4 **do for** each $v\in V:(u,v)\in E$
5 **do** $z_v=\min(z_v,z_u+\gamma_{uv})$

At the end of procedure DAGSP, z_{v_l} equals the warping distance between the time series $\mathbf{A}_i$ and $\mathbf{A}_k$, computed in $O(|V|)=O(T_iT_k)$ time. Therefore, the warping distances for all pairs of time series can be computed in $O(m^2T_{max}^2)$ time.

In practice, most papers performing warping distance computations have imposed some form of global or local constraints to prevent the alignment of periods positioned too far from each other along the time axis. It has been noticed [5] that these constraints not only decrease the time needed for evaluating the warping distances, but also lead to more accurate similarity metrics, improving the quality of the subsequent clustering or classification task. Nicely, it turns out that most constraints of this nature can be transposed into simple limitations on the network topology of the graph G.

For example, we will assume in the sequel a global constraint that permits to include a vertex $v=(r,s)$ in V only if $|r-s|\le\vartheta$, for a given integer parameter $\vartheta\ge 0$. This means that a vertex is in V only if the absolute difference between the periods r and s that can be aligned in the two time series is not greater than a fixed threshold ϑ. Due to this additional constraint, it follows that the graph G, denoted hereafter as G_ϑ, contains at most $[\vartheta\max(T_i,T_k)]$ vertices. Hence, the warping distance between each pair of time series can be computed by procedure DAGSP in $O(T_{max})$ time, leading to an overall $O(m^2T_{max})$ complexity for the whole set of warping distances. By letting ϑ vary along the time axis, one may also define more general constraints.

3. Rectangularization by fixed cardinality warping paths

As noticed in the Introduction, if the time series have variable length it is necessary to transform them into sequences of fixed length in order to achieve a rectangular table to be fed into a method for static classification. For instance, a rectangularization can be obtained by fixing a priori the number T of desired time periods, subdividing the time axis in T portions and then consolidating each attribute for each sequence over the time intervals by means of summaries or averages. Of course, this representation appears somewhat simplistic, and it is likely to loose important information embedded in the temporal dependence within each time series.

In order to derive a more effective rectangularization of the time series $\{\mathbf{A}_i\}$, capable of taking into account their profiles and capturing their behavior, in this section we propose a new

rectangularization scheme based on a fixed cardinality extension of the warping distance, in which the number H of arcs in the optimal warping path is fixed a priori and is constant for each pair of time series $(\mathbf{A}_i,\mathbf{A}_k)$. Hence, it is required to compute the shortest warping path in the graph G_ϑ subject to the additional constraint that the number of arcs in the path is equal to a constant value H. This problem is a single equality constrained version of the general resource constrained shortest path problem that has been investigated by different authors [18,19]. The single equality constrained version of the problem is $\mathcal{NP}$-*hard* for general graphs, since the traveling salesman problem can be reduced to it by taking $H=|V|-1$ and $v_f=v_l$. However, it can be solved in polynomial time if the graph is a DAG. In particular, we set $H=T_{max}$ since this choice leads to feasible solutions of the fixed cardinality warping distance problem, provided that the range in the time length of the m sequences is not too large.

Theorem 3.1. *Assuming $H=T_{max}$, for any pair of time series there exists at least a feasible warping path of cardinality H in the graph G_ϑ, provided the conditions $T_{max}<2T_{min}$ and $\vartheta\geq 1$ are satisfied.*

Proof. We know that for each pair of time series $\mathbf{A}_i$ and $\mathbf{A}_k$ the cardinality H of any warping path in G_ϑ lies in the interval $[\max(T_i,T_k),T_i+T_k-1]$, since $\vartheta\geq 1$. Hence, it is sufficient to show that the choice $H=T_{max}$ falls within this interval for any pair of time series. Indeed, by definition $H=T_{max}\geq\max(T_i,T_k)$. On the other hand, $T_i+T_k-1\geq 2T_{min}-1\geq T_{max}=H$. □

If the length of the time series has a large range of variability and the condition $T_{max}<2T_{min}$ is violated, there are at least two alternative ways to achieve the required regularity. First, one may drop the time series whose length falls on the tails of the length distribution, until the condition is met. Alternatively, to avoid a loss of discarded examples, one can apply a standard technique for replacing missing values in order to increase the value of T_{min}.

To compute the cardinality constrained shortest warping path in G_ϑ from the source vertex v_f to the destination vertex v_l, we need to modify as follows the procedure DAGSP:

procedure DAGSP-FC (G_ϑ,v_f,H)
1 topologically sort the vertices of G_ϑ
2 **for** each $h=1,2,\ldots,H$, **do** $\{z_{v_f}(h)=0$; **for** each $u\in V$ **do** $z_u(h)=\infty\}$
3 **for** each $u\in V$ in topological order
4 **do for** each $v\in V:(u,v)\in E$
5 **do for** $h=1,2,\ldots,H$
6 **do** $z_v(h)=\min(z_v(h),z_u(h-1)+\gamma_{uv})$

Theorem 3.2. *At the end of the procedure DAGSP-FC, $z_{v_l}(H)$ indicates the fixed cardinality warping distance between the time series $\mathbf{A}_i$ and $\mathbf{A}_k$, computed in $O(H|V|)=O(\vartheta T_{max}^2)=O(T_{max}^2)$ time for fixed ϑ.*

Proof. Let $\omega_v(h)$, $h=1,2,\ldots,H$, be the length of the shortest path from v_f to v containing exactly h arcs. We want to show that $z_v(h)=\omega_v(h)$ at termination of the algorithm, for each $v\in V$ and for each $h=1,2,\ldots,H$. The theorem is proved by induction on h.

For $h=1$, we have $z_v(1)=\gamma_{v_fv}=\omega_v(1)$ if $(v_f,v)\in E$, and $z_v(1)=\infty=\omega_v(1)$ if $(v_f,v)\notin E$.

We assume now that $z_v(h-1)=\omega_v(h-1)$ for each $v\in V$ and show that this implies $z_v(h)=\omega_v(h)$. By contradiction, suppose that $z_v(h)>\omega_v(h)$, and let $\{v_f,v_1,v_2,\ldots,v_{h-1},v\}$ be the sequence of vertices in the shortest path from v_f to v composed by h arcs. Since the vertices are topologically ordered, we have $\{v_f\prec v_1\prec v_2\prec\cdots\prec v_{h-1}\prec v\}$, where $\prec$ indicates the precedence relationship. Furthermore, since the vertices are considered in topological order in step 3, the values assumed by $z_u(j)$, $j=1,2,\ldots,H$ at the beginning of the step are retained until the end of the whole procedure. Consider the path $\{v_f,v_1,v_2,\ldots,v_{h-1}\}$, connecting v_f to v_{h-1} by means of $h-1$ arcs, and denote as $\phi_{v_{h-1}}(h-1)=\omega_v(h)-\gamma_{v_{h-1}v}$ its length. Due to the topological ordering, when vertex v is considered in step 4, the value $z_{v_{h-1}}(h-1)$ is permanently fixed. Consequently, it must be $z_v(h)\leq z_{v_{h-1}}(h-1)+\gamma_{v_{h-1}v}$. We have therefore

$$z_{v_{h-1}}(h-1)\geq z_v(h)-\gamma_{v_{h-1}v}>\omega_v(h)-\gamma_{v_{h-1}v}=\phi_{v_{h-1}}(h-1),$$

so that a path from v_f to v_{h-1} of cardinality $(h-1)$ and length strictly less than $z_{v_{h-1}}(h-1)$ has been found, contradicting the induction hypothesis. □

The rectangularization procedure is composed by the following main steps.

1. First, we compute the warping distances e_{ik}, $i,k\in\mathcal{M},i<k$, between all pairs of time series, by repeatedly using the procedure DAGSP. Then, for each class label $d\in\mathcal{D}$ we calculate the *centroid* $\mathbf{A}_c$ of the class as the time series for which the sum of the warping distances to all other time series in the class is minimum, that is
$$\mathbf{A}_c=\underset{\mathbf{A}_i:y_i=d}{\operatorname{argmin}}\sum_{\substack{\mathbf{A}_k:y_k=d\\ i\neq k}}e_{ik}.$$
2. For each time series $\mathbf{A}_i$, $i\neq c$, having the class label $y_i=d$ we compute the fixed cardinality warping distance between $\mathbf{A}_c$ and $\mathbf{A}_i$, by means of the procedure DAGSP-FC.
3. We then build a three-dimensional $m\times L\times H$ matrix in which the first entry corresponds to the time series, the second to the attributes and the third to the number H of arcs in the optimal warping path. For each series $\mathbf{A}_i$ we take its value in the position (i,l,h) of the matrix as the value a_{ilr}, where r is the time period aligned in the h-th arc of the warping path connecting the time series $\mathbf{A}_i$ to the corresponding centroid $\mathbf{A}_c$.
4. Finally, to obtain a rectangular $m\times n$ matrix, with $n=L\times H$, we proceed by sequencing for each time series $\mathbf{A}_i$ the attributes and the time periods.

Based on the previous discussion, it can be readily seen that the whole rectangularization procedure leads to an overall $O(m^2T_{max}^2)$ time complexity.

4. Temporal discrete support vector machines

At the end of the rectangularization phase described in Section 3, we may assume that the set of time series is represented by a $m\times n$ matrix. In this section, we propose a new optimization problem which extends the notion of discrete SVM in order to perform time series classification. Here, we confine the attention to binary classification tasks. General multicategory classification problems can be reduced to sequences of binary tasks by means of *one-against-all* or *all-against-all* schemes [20,13].

For many binary classification methods the generic hypothesis takes the form $f(\mathbf{x})=\operatorname{sgn}(g(\mathbf{x}))$, where $g:\Re^n\mapsto\Re$ is a properly defined *score function*. If the space $\mathcal{H}$ is based on the set of separating hyperplanes in $\Re^n$, we have $g(\mathbf{x})=\mathbf{w}'\mathbf{x}-b$. In order to choose the optimal parameters $\mathbf{w}$ and b, SVM [22–24] resort to the minimization of the following risk functional:

$$R(f)=\frac{1}{m}L(y,f(\mathbf{x}))+\lambda\|f\|_K^2, \tag{4}$$

where $K(\cdot,\cdot)$ is a given symmetric positive definite function named *kernel*; $\|f\|_K^2$ denotes the norm of f in the reproducing kernel

Hilbert space induced by K [21] and plays a regularization role; $L(yf(\mathbf{x}))$ is a loss function that measures the accuracy by which the predicted output $f(\mathbf{x})$ approximates the actual output y; λ is a parameter that controls the trade-off between the empirical error and the regularization term.

In the classical theory of SVM [22–24] the loss function measures the distance of the misclassified examples from the separating hyperplane. *Discrete* SVM represent a variant of SVM, introduced in [11,12], for which the loss is expressed by a discrete function which counts the number of misclassified examples. The rational behind discrete SVM is that a precise evaluation of the empirical error may determine a more accurate classifier. This leads to the formulation of a mixed-integer optimization problem that corresponds to the minimization of (4) using the discrete loss function, with the inclusion of an additional regularization term representing the number of attributes which define the separating hyperplane. This term is aimed at reducing the dimension of the space $\mathcal{H}$, in order to derive optimal hypotheses of lower complexity and higher generalization capability.

In the discrete SVM framework, the number of misclassified examples is computed by means of the binary variables

$$p_i = \begin{cases} 0 & \text{if } \mathbf{x}_i \text{ is correctly classified,} \\ 1 & \text{if } \mathbf{x}_i \text{ is misclassified,} \end{cases} \tag{5}$$

while the count of the number of attributes defining the separating hyperplane is based on the binary variables

$$q_j = \begin{cases} 0 & \text{if } w_j = 0, \\ 1 & \text{if } w_j \neq 0. \end{cases} \tag{6}$$

Let $c_i, i \in \mathcal{M}$, be a penalty for the misclassification of example $\mathbf{x}_i$, and $h_j, j \in \mathcal{N}$, the penalty cost of using attribute j. Let also S and R be sufficiently large constant values, and α,β,γ the parameters to control the trade-off among the objective function terms. The problem of determining an optimal separating hyperplane is formulated through the following *discrete support vector machines* model

$$\min \quad \frac{\alpha}{m}\sum_{i=1}^{m} c_i p_i + \frac{\beta}{2}\sum_{j=1}^{n} u_j + \frac{\gamma}{n}\sum_{j=1}^{n} h_j q_j \tag{DSVM}$$

$$\text{s.t.} \quad y_i(\mathbf{w}'\mathbf{x}_i - b) \geq 1 - S p_i \qquad i \in \mathcal{M} \tag{7}$$

$$u_j \leq R q_j, \quad j \in \mathcal{N} \tag{8}$$

$$-u_j \leq w_j \leq u_j, \quad j \in \mathcal{N} \tag{9}$$

$$\mathbf{u} \geq \mathbf{0}, \quad \mathbf{p},\mathbf{q} \text{ binaries}, \quad \mathbf{w}, b \text{ free}.$$

The family of bounding variables $u_j, j \in \mathcal{N}$, and the constraints (9) are introduced in order to linearize the norm of f in the risk functional (4).

Although model (DSVM) is directly applicable to a dataset obtained from the rectangularization of the time series, we propose an extension of the original discrete SVM formulation by defining two new regularization terms aimed at improving the discrimination capability when dealing with temporal classification problems.

In order to clarify the role of the first regularizer, observe that the loss functions used in SVM and discrete SVM stick on the assumption of leaving misclassified the examples lying inside the strip region determined by the *margin of separation*, defined as the distance between the pair of parallel *canonical supporting hyperplanes* $\mathbf{w}'\mathbf{x} - b - 1 = 0$ and $\mathbf{w}'\mathbf{x} - b + 1 = 0$. In model (DSVM) the width of the margin is regulated by the second term in the objective function, which expresses the linear representation of the norm of f. To improve the discrimination among time series of distinct classes, we adopt an alternative perspective for which the margin is explicitly determined by the inclusion of a new variable $\varepsilon > 0$, and the discrete loss function is modified in order to account only for those sequences which are actually misclassified by the optimal hypothesis f^*.

The second regularization term is represented by the sum of the warping distances between all pairs of time series assigned to the same class. Since the warping distance can be retained as a similarity measure, the inclusion of this term into the objective function aims at determining a separating hyperplane which is optimal also with respect to time series likeness. Indeed, it is reasonable to assume that temporal sequences belonging to the same class usually exhibit resemblance in their temporal profiles.

Let each example $\mathbf{x}_i$ represent the row corresponding to time series $\mathbf{A}_i$ in the rectangular representation obtained at the end of the first phase. Let e_{ik} denote the warping distance between the pair of temporal sequences $(\mathbf{x}_i,\mathbf{x}_k)$ of the training set $\mathcal{S}_m$. In order to determine the best separating function for time series classification the following nonsmooth optimization problem can be formulated:

$$\min \quad \frac{1}{m}\sum_{i=1}^{m} c_i p_i + \frac{1}{2}\sum_{j=1}^{n} u_j + \frac{2\delta}{m(m-1)}\sum_{i=1}^{m}\sum_{k=i+1}^{m} e_{ik} \times \frac{|y_i(2p_i-1)+y_k(2p_k-1)|}{2} - v\varepsilon \qquad \text{(NTDVM)}$$

$$\text{s.t.} \quad y_i(\mathbf{w}'\mathbf{x}_i - b) \geq \varepsilon - S p_i \qquad i \in \mathcal{M} \tag{10}$$

$$-u_j \leq w_j \leq u_j, \quad j \in \mathcal{N} \tag{11}$$

$$\mathbf{u} \geq \mathbf{0}, \ \varepsilon \geq \rho, \ \mathbf{p} \text{ binary}, \quad \mathbf{w}, b \text{ free}.$$

Here $\rho > 0$ is a lower threshold to prevent the case $\varepsilon = 0$ that might lead to useless optimal solutions for which $\|f\| = 0$. In model (NTDVM) the variable ε plays a regularization role since it progressively softens or hardens the separation between the two classes determined by the optimal hyperplane. When ε decreases and approaches 0 the margin around the separating hyperplane reduces and tends to vanish, whereas the opposite is true when ε increases. Notice that the explicit inclusion of the variable ε allows to fix to 1 the values of the parameters α and β in the objective function, since the trade-off between the misclassification error and the generalization capability of the classifier is regulated by the parameter v. In particular, when v is large there is an advantage in taking ε large as well, and the misclassification error increases. By converse, small values of v induce ε to decrease, reducing the empirical error at the expense of the generalization capability.

Theorem 4.1. *Given a feasible solution to problem* (NTDVM), *the expression*

$$\sum_{i=1}^{m}\sum_{k=i+1}^{m} e_{ik}\frac{|y_i(2p_i-1)+y_k(2p_k-1)|}{2} \tag{12}$$

is equal to the sum of the warping distances between all pairs of time series assigned to the same class.

Proof. Let $s_i = y_i(2p_i - 1)$, $\forall i \in \mathcal{M}$. Consider first two examples, $\mathbf{x}_i$ and $\mathbf{x}_k$, belonging to the class $\{+1\}$, so that $y_i = y_k = 1$. If $\mathbf{x}_i$ and $\mathbf{x}_k$ are assigned to the positive class, they are both correctly classified, and the binary variables p_i and p_k are forced to take the value 0 by constraints (10). In this case, $s_i = s_k = -1$ and the warping distance between $\mathbf{x}_i$ and $\mathbf{x}_k$ is correctly computed in (12). On the contrary, if $\mathbf{x}_i$ and $\mathbf{x}_k$ are labeled with the negative class, they result in two misclassified examples, and constraints (10)

force to 1 the binary variables p_i and p_k. In this case, $s_i=s_k=1$ and the warping distance between the two examples is still accounted in the sum (12), as required. Finally, if $\mathbf{x}_i$ and $\mathbf{x}_k$ are assigned to opposite classes, the variable s takes the value -1 for the correctly classified example and the value 1 for the misclassified one, and the warping distance between $\mathbf{x}_i$ and $\mathbf{x}_k$ is not included in (12). An analogous reasoning can be developed when the examples $\mathbf{x}_i$ and $\mathbf{x}_k$ belong to the class $\{-1\}$. □

The presence of the two new regularization terms in (NTDVM) led us to drop the third term appearing in the objective function of model (DSVM), aimed at controlling the complexity of the optimal hypothesis f^*, since the computational experiences indicated that its role became practically irrelevant.

The inclusion of the sum of the warping distances between the temporal sequences, expressed by (12), leads to a non-differentiable objective function in the mixed-integer optimization problem. In order to reformulate model (NTDVM) as a linear problem, we define the family of continuous bounding variables $r_{ik}, i,k \in \mathcal{M}$, and obtain the following optimization problem termed *temporal discrete support vector machines* (TDVM):

$$\min \quad \frac{1}{m}\sum_{i=1}^{m} c_i p_i + \frac{1}{2}\sum_{j=1}^{n} u_j + \frac{\delta}{m(m-1)}\sum_{i=1}^{m}\sum_{k=i+1}^{m} e_{ik} r_{ik} - v\varepsilon \qquad \text{(TDVM)}$$

$$\text{s.t.} \quad y_i(\mathbf{w}'\mathbf{x}_i - b) \geq \varepsilon - S p_i, \quad i \in \mathcal{M} \tag{13}$$

$$-r_{ik} \leq y_i(2p_i-1) + y_k(2p_k-1) \leq r_{ik}, \quad i,k \in \mathcal{M}, i<k \tag{14}$$

$$-u_j \leq w_j \leq u_j, \quad j \in \mathcal{N} \tag{15}$$

$$\mathbf{u} \geq \mathbf{0}, \ r_{ik} \geq \mathbf{0} \quad i,k \in \mathcal{M} \ \ \varepsilon \geq \rho, \ \mathbf{p} \text{ binary}, \quad \mathbf{w}, b \text{ free}.$$

For determining a feasible suboptimal solution to model (TDVM), we adopt a heuristic procedure based on a sequence of linear programming (LP) problems. The heuristic starts by considering the LP relaxation of problem (TDVM). Each LP problem $(\text{TDVM})_{t+1}$ in the sequence is obtained by fixing to zero the relaxed binary variable with the smallest fractional value in the optimal solution of the predecessor $(\text{TDVM})_t$. The procedure is stopped if problem $(\text{TDVM})_t$ is feasible and its optimal solution is integer feasible, and the solution generated at iteration t is retained as an approximation to the optimal solution of problem (TDVM). Otherwise, if problem $(\text{TDVM})_{t+1}$ is unfeasible, the procedure modifies the previous LP problem $(\text{TDVM})_t$ by fixing to 1 all of its fractional variables. Problem $(\text{TDVM})_{t+1}$ defined in this way is feasible and any of its optimal solutions is integer. Thus, the procedure is stopped and the solution found for $(\text{TDVM})_t$ is retained as an approximation to the optimal solution of (TDVM).

5. Computational setup and analysis

The proposed method, which combines the rectangularization stage with the classification task performed by model (TDVM), has been evaluated on six datasets mostly composed by multivariate temporal sequences. Four of these datasets are usually considered as benchmark datasets for comparing the accuracy of alternative classification methods. These are "Japanese vowels" (*Jvowels*), "robot execution failures" (*Robot*), "pen-based recognition of handwritten digits" (*Pendigits*), all available from the UCI KDD Archive [25], and *ECG* available at the UCR Time Series Homepage [26]. The last two datasets, indicated as *Telecom* and *Electronics*, are real world marketing datasets referring, respectively, to a retention analysis for a telecommunication company and a cross-selling application in the consumer electronics industry.

In particular, *Jvowels* consists of multivariate time series of variable length given by the utterances of two Japanese vowels provided by nine male speakers. *Robot* refers to five classification tasks concerning the identification of a robot execution failures. Each learning problem is based on a sample of multivariate sequences of fixed length, where each series is described in terms of numerical attributes providing the force and torque measurements collected after failure detection. *Pendigits* consists of bivariate time series of fixed length representing handwriting digits samples. *ECG* is composed by univariate sequences of fixed length reporting the measurements of the cardiac activity as recorded by one electrode during one heartbeat. Finally, *Telecom* and *Electronics* contain sequences of variable length described both by numerical and categorical attributes. In the first case, each sequence refers to a customer who has churned or who is still loyal. In the second case, each series corresponds to an individual who, in the past, has adhered or not to a digital camera purchase promotion. For the marketing datasets the problem is to devise an optimal classification function for identifying segments of customers who are more likely to churn or to accept a future promotional offer. The distinctive features of each dataset in terms of number of classes, variables, time series length and number of available examples, are summarized in Table 1. These datasets were selected because they are illustrative of a wide range of temporal classification problems.

Six alternative methods were considered for comparisons with temporal discrete SVM (TDVM): discrete SVM (DSVM), SVM with linear (SVM_{LIN}), sigmoid (SVM_{SIG}), radial basis function (SVM_{RBF}) and dynamic time warping (SVM_{DTW}) kernels, and the 1-nearest neighbor classifier (1NN_{WD}) based on the warping distance described in Section 2. In particular, this latter appeared as one of the most robust and accurate classifier for time series in extensive computational tests [6]. Among the kernels based on dynamic time warping, we implemented the one proposed in [9] since it is positive definite under favorable conditions. The results for classifiers (TDVM) and (DSVM) were derived using the heuristic procedure described in Section 4, whereas for SVM the LIBSVM library [27] was employed, extending its standard version with the dynamic time warping kernel. In order to perform the multicategory classification of *Jvowels*, *Pendigits* and *Robot*, models (TDVM) and (DSVM) were framed within the all-against-all scheme described in [13]. Moreover, in applying all the classifiers each categorical attribute in *Telecom* and *Electronics* was converted into a numerical explanatory variable. This was achieved by replacing each categorical value with the conditional probability of observing that value given the positive class. The rational behind this encoding is to replace each category with a value that takes into account its individual relevance on the target class.

Table 1
Description of the temporal datasets.

Dataset	Classes	Variables	Length	Examples
Jvowels	9	12	[7…29]	640
Pendigits	10	2	8	10 992
ECG	2	1	96	200
Telecom	2	32	[18…24]	1200
Electronics	2	27	[35…48]	1800
Robot				
Failure 1	4	6	15	88
Failure 2	5	6	15	47
Failure 3	4	6	15	47
Failure 4	3	6	15	117
Failure 5	5	6	15	164

Table 2
Classification accuracy (%) and 95% confidence intervals (%) on the temporal datasets.

Dataset	**Method**						
	TDVM	DSVM	SVM_{LIN}	SVM_{SIG}	SVM_{RBF}	SVM_{DTW}	$1NN_{WD}(\vartheta)$
Jvowels	96.6 $\pm$0.010	93.6 $\pm$0.018	96.3 $\pm$0.011	**96.7** $\pm$0.010	96.6 $\pm$0.010	94.6 $\pm$0.016	92.3 (9) $\pm$0.022
Pendigits	96.3 $\pm$0.001	94.6 $\pm$0.001	94.8 $\pm$0.001	94.7 $\pm$0.001	**97.2** $\pm$0.000	93.4 $\pm$0.001	94.5 (1) $\pm$0.001
ECG	**90.5** $\pm$0.084	85.5 $\pm$0.121	84.5 $\pm$0.128	83.5 $\pm$0.135	89.5 $\pm$0.092	86.0 $\pm$0.118	90.0 (1) $\pm$0.088
Telecom	**95.2** $\pm$0.007	92.4 $\pm$0.011	86.3 $\pm$0.019	86.2 $\pm$0.019	88.3 $\pm$0.017	84.3 $\pm$0.022	86.2 (3) $\pm$0.019
Electronics	**85.8** $\pm$0.013	82.7 $\pm$0.016	80.1 $\pm$0.017	79.7 $\pm$0.018	82.4 $\pm$0.016	78.2 $\pm$0.019	80.2 (2) $\pm$0.017
Robot							
Failure 1	**85.2** $\pm$0.281	72.7 $\pm$0.442	71.6 $\pm$0.453	70.5 $\pm$0.463	83.0 $\pm$0.314	81.8 $\pm$0.332	81.8 (3) $\pm$0.332
Failure 2	**63.8** $\pm$0.963	57.4 $\pm$1.020	55.3 $\pm$1.031	55.3 $\pm$1.031	57.4 $\pm$1.020	**63.8** $\pm$0.963	59.6 (3) $\pm$1.004
Failure 3	**68.1** $\pm$0.906	63.8 $\pm$0.963	55.3 $\pm$1.031	53.2 $\pm$1.038	66.0 $\pm$0.936	65.8 $\pm$0.938	61.7 (1) $\pm$0.985
Failure 4	85.5 $\pm$0.208	75.2 $\pm$0.312	77.8 $\pm$0.289	83.6 $\pm$0.230	84.6 $\pm$0.218	**87.2** $\pm$0.187	86.3 (3) $\pm$0.198
Failure 5	**67.1** $\pm$0.264	56.7 $\pm$0.293	54.3 $\pm$0.297	50.6 $\pm$0.299	55.5 $\pm$0.295	62.1 $\pm$0.281	65.2 (2) $\pm$0.271

The performance of the competing classifiers was estimated according to the size of each dataset. Due to the restricted number of available examples, for *ECG* and the robot failure datasets 5-fold cross-validation was applied. For *Jvowels*, *Pendigits*, *Telecom* and *Electronics* we used instead 10-fold cross-validation. To the aim of investigating the effect of the modeling parameters, different settings were setup regarding the value of ϑ, the weight factors $\alpha, \beta, \gamma, \delta$ and v appearing in the formulation of models (DSVM) and (TDVM) and the kernel parameters in SVM methods. The most promising values of the parameters were empirically found for each classifier and for each training set by means of successive refinements of a grid search. In particular, the parameter ϑ took its value in the range [1,10] with step 1. For discrete SVM methods, models were generated by varying α, β, γ, δ and v in the interval [0,1] with step 0.2. For SVM classifiers we considered each combination of the regularization constant $C=\{10^i, i=-1,\ldots,4\}$ with the kernels parameters. Specifically, the scaling and the shifting parameters of the sigmoid kernel ranged, respectively, in the intervals [0.0001,1] and $[-3,-1]$ with steps of variable length; the RBF and the DTW kernel parameters were varied, respectively, in the ranges [0.01,10] and [0.1,30] again with variable grid steps. Finally, the classification accuracy and the confidence interval were estimated as suggested in [28] on all datasets.

The results presented in Table 2 indicate that the proposed method has a great potential to perform accurate time series classification. On most datasets considered in our tests, temporal discrete SVM provided the highest rate of correct predictions, leading to an increase in the accuracy ranging between 0.5% and 3.1%. The most significant improvement is achieved for multivariate datasets. However, the performance exhibited on *ECG* deserves attention too, since it shows that model (TDVM) can lead to accurate predictions also for univariate temporal sequences.

It is worth to notice that the training and the classification were quite fast for all datasets, requiring only a few seconds for computing offline the warping distances, and then a time ranging from seconds to less than a minute for applying the approximate algorithm described in Section 4.

The general improvement in accuracy achieved by model (TDVM) regards in the same way datasets containing fixed and variable length time series, and therefore is not tied to the rectangularization method based on the fixed cardinality warping distance. However, the rectangularization technique proposed seems to be in itself beneficial. To show this empirically, we applied to *Telecom* and *Electronics* an alternative rectangularization mechanism based on averaging over a fixed number $(T_{min}+T_{max})/2$ of time periods. The improvement in accuracy achieved on the same datasets rectangularized by the fixed cardinality warping distance method ranged between 0.5% and 1.4%, consistently for the different classifiers tested, as shown in Table 3.

Table 3
Classification accuracy (%) by averaging over a fixed number of time periods.

Dataset	**Method**						
	TDVM	DSVM	SVM_{LIN}	SVM_{SIG}	SVM_{RBF}	SVM_{DTW}	$1NN_{WD}$
Telecom	94.4	91.2	85.2	85.5	87.7	82.9	85.6
Electronics	85.1	81.8	79.3	79.1	81.9	77.7	79.4

Finally, Table 2 suggests that temporal discrete SVM represent an improvement with respect to model (DSVM) when dealing with the classification of multivariate time series.

6. Conclusions

In this paper we have proposed a new framework for the classification of multivariate time series of variable length composed by a two-phase procedure. In the first phase, time series are converted into sequences of the same length by means of a new rectangularization technique based on a fixed cardinality version of the warping distance, which represents a robust and versatile similarity measure between pairs of time series. In the second phase, for addressing the classification task we have developed a temporal variant of discrete SVM which incorporates two regularization terms: the first term allows to control more effectively the trade-off between accuracy and potential of generalization, by including the margin of separation as a variable into the optimization model. The second term is given by the sum

of the warping distances between the pairs of time series assigned to the same class, and is aimed at determining a separating hyperplane which is optimal also with respect to time series similarity. The effectiveness of the proposed framework has been evaluated on four benchmark datasets and on two real-world marketing datasets. On most datasets temporal discrete SVM outperformed traditional SVM and the 1-nearest neighbor classifier, leading to an increase in the accuracy ranging between 0.5% and 3.1%. Also the rectangularization procedure seemed to be beneficial, since it provided an improvement in accuracy ranging between 0.5% and 1.4% when compared with standard rectangularization techniques based on averaging over a fixed number of time periods. Future extensions of the proposed approach will be pursued along two main directions, concerning, respectively, the evaluation of alternative similarity measures among time series and the use of nonlinear kernels in the temporal discrete SVM classification model.

References

[1] M.W. Kadous, C. Sammut, Classification of multivariate time series and structured data using constructive induction, Machine Learning 58 (2005) 179–216.

[2] Y. Wu, E.Y. Chang, Distance-function design and fusion for sequence data, in: Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management, 2004, pp. 324–333.

[3] A. Nanopoulos, R. Alcock, Y. Manolopoulos, Feature-based classification of time-series data, International Journal of Computer Research 10 (2001) 49–61.

[4] J.J. Rodriguez, C.J. Alonso, Interval and dynamic time warping-based decision trees, in: Proceedings of the 2004 ACM Symposium on Applied Computing, 2004, pp. 548–552.

[5] E. Keogh, C.A. Ratanamahatana, Exact indexing of dynamic time warping, Knowledge and Information Systems 7 (2004) 358–386.

[6] X. Xi, E. Keogh, C. Shelton, L. Wei, Fast time series classification using numerosity reduction, in: Proceedings of the 23nd International Conference on Machine Learning, 2006, pp. 1033–1040.

[7] H. Shimodaira, K. Noma, M. Nakai, S. Sagayama, Support vector machine with dynamic time-alignment kernel for speech recognition, in: Proceedings of Eurospeech, 2001, pp. 1841–1844.

[8] C. Bahlmann, B. Haasdonk, H. Burkhardt, On-line handwriting recognition with support vector machines: a kernel approach, in: Frontiers in Handwriting Recognition, 2002, pp. 49–54.

[9] M. Cuturi, J.P. Vert, O. Birkenes, T. Matsui, A kernel for time series based on global alignments, in: Proceedings of ICASSP, 2007, pp. 413–416.

[10] W. Chaovalitwongse, P.M. Pardalos, On the time series support vector machine using dynamic time warping kernel for brain activity classification, Cybernetics and Systems Analysis 44 (2008) 125–138.

[11] C. Orsenigo, C. Vercellis, Multivariate classification trees based on minimum features discrete support vector machines, IMA Journal of Management Mathematics 14 (2003) 221–234.

[12] C. Orsenigo, C. Vercellis, Discrete support vector decision trees via tabu-search, Journal of Computational Statistics and Data Analysis 47 (2004) 311–322.

[13] C. Orsenigo, C. Vercellis, Multicategory classification via discrete support vector machines, Computational Management Science 6 (2009) 101–114.

[14] C. Orsenigo, C. Vercellis, Accurately learning from few examples with a polyhedral classifier, Computational Optimization and Applications 38 (2007) 235–247.

[15] H. Sakoe, C. Chiba, Dynamic programming algorithm optimization for spoken word recognition, IEEE Transactions on Acoustics, Speech, and Signal Processing 26 (1978) 43–49.

[16] M. Vlachos, M. Hadjieleftheriou, D. Gunopulos, E. Keogh, Indexing multidimensional time-series, The VLDB Journal 15 (2006) 1–20.

[17] R. Sedgewick, Algorithms in Java, Part 5: Graph Algorithms, Addison-Wesley, Boston, 2003.

[18] J. Beasley, N. Christofides, An algorithm for the resource constrained shortest path problem, Networks 19 (1989) 379–394.

[19] S. Irnich, G. Desaulniers, Shortest path problems with resource constraints, in: G. Desaulniers, J. Desrosiers, M. Solomon (Eds.), Column Generation, Springer, 2005, pp. 33–65.

[20] E. Allwein, R. Schapire, Y. Singer, Reducing multiclass to binary: a unifying approach for margin classifiers, Journal of Machine Learning Research 1 (2000) 113–141.

[21] C. Berg, J.P.R. Christensen, P. Ressel, Harmonic Analysis on Semigroups: Theory of Positive Definite and Related Functions, Springer, New York, 1984.

[22] V. Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag, New York, 1995.

[23] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods, Cambridge University Press, Cambridge, 2000.

[24] B. Schölkopf, A.J. Smola, Learning with Kernels. Support Vector Machines, Regularization, Optimization and Beyond, MIT Press, Cambridge, 2002.

[25] S. Hettich, S. Bay, The UCI KDD Archive, 1999 〈http://kdd.ics.uci.edu〉.

[26] E. Keogh, X. Xi, L. Wei, C. Ratanamahatana, The UCR Time Series Classification/Clustering Homepage, 2006 〈http://www.cs.ucr.edu/∼eamonn/time_series_data/〉.

[27] C.C. Chang, C.J. Lin, LIBSVM: a library for support vector machines, 2001.

[28] R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, in: Proceedings of the International Joint Conference on Artificial Intelligence, 1995, pp. 1137–1143.

Carlotta Orsenigo is an assistant professor of Operations Research at the School of Management of Politecnico di Milano, where she teaches courses in Optimization and Data Mining. Her current research interests include mathematical models and methods for machine learning and pattern recognition and their application to marketing, social network analysis and life sciences.

Carlo Vercellis is full professor of Operations Research at the School of Management of Politecnico di Milano, where he teaches courses in Optimization, Business Intelligence and Data Mining. He is director of the research group MOLD—mathematical modeling, optimization, learning from data. His current research interests include mathematical models for data mining and machine learning, such as support vector machines; optimization models, with applications to supply chain and revenue management. In the past he was involved in research on design and analysis of algorithms for combinatorial optimization, project management, transportation models.