

Time series classification by discrete support vector machines

Carlotta Orsenigo*, Carlo Verzellis†

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 made by a fixed number of attributes. Then, any of the existing efficient methods for classification can be applied for learning and predicting the class of future temporal sequences.

In this paper we propose an extension of discrete support vector machines, that have been shown to outperform other competing classification methods on benchmark datasets, for time series classification. In order to transform a temporal dataset into the rectangular shape we also develop a constrained variant of dynamic time warping. Preliminary computational results on marketing datasets indicate the effectiveness of the proposed method in comparison to other techniques.

Keywords: Classification, Time series, Temporal data mining, Learning theory

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. For instance, a large majority of classification problems arising in the field of relational marketing are based on sequential data: the behavior of customers is observed through time, and their buying attitude or their interactions with the company do certainly compose multivariate time series. When dealing with marketing data, it is a common practice to reduce them to tabular shapes by simply consolidating the variables along vertical time frames on the temporal dimension. We believe that by properly framing classification problems in the marketing field within a temporal setting may lead to a higher forecasting accuracy, as shown by our computational experience.

In general, several alternative paradigms for time series classification have been proposed, that we cannot summarize due to space limitations; we refer the reader to (Kadous and Sammut, 2005). However, we observe that a common approach is based on a two-step 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 attributes, through different rectangularization mechanisms; then, a static classification method is applied for labeling the data, such as support vector machines (Wu and Chang, 2004), neural networks (Nanopoulos *et al.*, 2001), induction trees (Rodriguez and Alonso, 2004), and so forth. Another approach, which seems to be very effective for univariate time series classification, is instead based on the definition of a suitable measure of similarity between pairs of time series allowing to detect clusters as a mean for predicting the class of new temporal sequences (Xi *et al.*, 2006; Keogh and Ratanamahatana, 2004).

*Dipartimento di Scienze Economiche, Aziendali e Statistiche, Università di Milano, Via Conservatorio 7, 20122 Milano, Italy. *E-mail address:* carlotta.orsenigo@unimi.it

†Corresponding author. Dipartimento di Ingegneria Gestionale, Politecnico di Milano, P.za Leonardo da Vinci 32, 20133 Milano, Italy. *E-mail address:* carlo.verzellis@polimi.it

In this paper, we are aimed at applying a family of methods known as discrete support vector machines (DSVM) to time series classification. Different variants of DSVM have been shown to outperform other techniques when dealing with classification problems (Orsenigo and Vercellis, 2003, 2004, 2006a,b). In order to fully exploit the intrinsic temporal dependence in the data, we propose a modified dynamic time warping (DTW) method to derive a rectangular representation of time series. By this way, the dependence from time is preserved in the derived tabular shape, and furthermore a proper phasing and alignment of the time series is fully exploited. 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. Furthermore, we propose a new temporal variant of DSVM in which the optimization model takes into account also the total similarity among time series belonging to the same class.

The paper is organized as follows. Section 2 defines the time series classification problem and describes the proposed rectangularization technique based on constrained DTW. In section 3 a new classification model based on temporal DSVM is presented. Finally, in section 4 computational setup and experiences are discussed.

2 Rectangularization by constrained dynamic time warping

In classification problems, termed *static* to underline the difference from *temporal* classification defined below, a set $\mathcal{S}_m = \{(\mathbf{x}_i, y_i), i \in \mathcal{M} = \{1, 2, \dots, m\}\}$ of training input-output examples is given. Here $\mathbf{x}_i \in \mathbb{R}^n$ is an input vector of attributes and $y_i \in \mathcal{D} = \{1, 2, \dots, D\}$ is the categorical output value associated to \mathbf{x}_i . Each component x_{ij} of an example \mathbf{x}_i is a realization of a random variable $B_j, j \in \mathcal{N} = \{1, 2, \dots, n\}$, that will be referred to as an attribute of \mathcal{S}_m . Let \mathcal{H} denote a set of functions $f(\mathbf{x}) : \mathbb{R}^n \mapsto \mathcal{D}$ that represent hypothetical relationships between \mathbf{x}_i and y_i . A 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 refer to *binary* classification problems, while the general case is termed *multicategory* classification. For binary problems we assume that $y_i \in \{-1, 1\}$, without loss of generality.

In temporal classification problems 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 $l \in \mathcal{L} = \{1, 2, \dots, L\}$ is the index associated to the attributes of the time series, whereas $t \in \mathcal{T}_i = \{1, 2, \dots, T_i\}$ is the temporal index, that may vary in a different range for each \mathbf{A}_i . To every time series is also associated a class label $y_i \in \mathcal{D}$. The 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.

As an example of temporal classification, consider a relational marketing problem in the telecommunication industry where each time series \mathbf{A}_i corresponds to the transactions of a single customer recorded at different time periods, and the attributes may represent the number, value and duration of the calls made for different types of connections. To model and predict customers loyalty, one can formulate a binary classification problem in which the label y_i indicates whether a customer is still active (+1) or has *churned* (-1), presumably leaving the company for accessing the services of some competitor.

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, it is rather natural to develop a two-phase procedure for dealing with time series classification. 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 obtained in the first phase.

For instance, to achieve a rectangularization one may fix a priori the number T of desired time periods, subdividing the time axis in T portions and then consolidate by summing or averaging each attribute of each series over the different time intervals. Of course, this representation appears somewhat simplistic, and it is likely to loose important information embedded in temporal dependence within each time series.

In this section we propose a rectangularization scheme based on a constrained extension of dynamic time warping (DTW), a method originally proposed in the context of speech recognition and signal processing (Sakoe and Chiba (1978)) and successfully applied as a proximity measure for clustering and labeling univariate (1-dimensional) and bivariate (planar) time series (Xi *et al.* (2006); Vlachos *et al.* (2006)). As a

similarity measure, DTW has proven to be more robust and versatile than the Euclidean distance 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 DTW distance can be calculated in polynomial time for the 1-dimensional case, whereas its computation becomes \mathcal{NP} -complete for $L \geq 2$.

We start by describing DTW for univariate time series, that is $L = 1$, where a single attribute is recorded for each time series along its time trajectory. Given two univariate time series \mathbf{A}_i and \mathbf{A}_k , let $\mathbf{G}_{ik} = [g_{ik}(r, s)]$, $r = 1, 2, \dots, T_i$, $s = 1, 2, \dots, T_k$, be a $T_i \times T_k$ matrix whose generic element $g_{ik}(r, s)$ corresponds to the squared distance obtained by the potential alignment of periods r and s in the two series, i.e.

$$g_{ik}(r, s) = (a_{i1r} - a_{k1s})^2. \quad (1)$$

In order to find the optimal alignment between \mathbf{A}_i and \mathbf{A}_k , we have to compute a *warping path* through the matrix \mathbf{G}_{ik} , that is a contiguous path starting from the bottom leftmost corner $g_{ik}(1, 1)$, ending at the top rightmost corner $g_{ik}(T_i, T_k)$ and minimizing the total cumulative distance. Hence, a warping path $P_{ik} = \{(r_1, s_1), (r_2, s_2), \dots, (r_V, s_V)\}$ is represented by a sequence of V pairs of time periods, corresponding to a path of contiguous cells in the matrix \mathbf{G}_{ik} . It can be easily seen that the warping path determines also a bipartite matching between the time periods of the two time series. In general, the length V of the sequence is variable and lies in the interval $[\max(T_i, T_k), T_i + T_k - 1]$.

The warping path can be computed by a dynamic programming algorithm, based on the following recursive equation

$$q(r, s) = g_{ik}(r, s) + \min\{q(r-1, s-1), q(r-1, s), q(r, s-1)\}, \quad (2)$$

where $q(r, s)$ denotes the cumulative distance of the path from cell $g_{ik}(1, 1)$ to cell $g_{ik}(r, s)$. In practice, the number of paths considered during the search for the warping path can be reduced by imposing a number of constraints (Keogh and Ratanamahatana, 2004).

Turning to multivariate time series, the concept of warping path can be generalized by defining the distance associated to the potential alignment of periods r and s in the two series \mathbf{A}_i and \mathbf{A}_k as

$$g_{ik}(r, s) = \sum_{l=1}^L (a_{ilr} - a_{kls})^2. \quad (3)$$

By a natural extension of the recursive equation, one can derive a dynamic programming algorithm also for the multidimensional case $L \geq 2$, although at the expense of a higher time complexity. However, notice that the dynamic programming algorithm can be turned into an effective heuristic procedure that approximates the warping path by applying suitable constraints that limit the search space.

In order to derive a rectangularization of the time series $\{\mathbf{A}_i\}$, capable of taking into account their profiles and capturing their behavior, we consider a constrained extension of DTW in which the number V of cells in the warping path is fixed a priori and is constant for each pair of time series $(\mathbf{A}_i, \mathbf{A}_k)$. Actually, we select from the training set a template time series $\mathbf{A}_{i'}$ for each class label and we compute the constrained warping path with constant V between $\mathbf{A}_{i'}$ and each other time series belonging to the same class. This leads to $O(m)$ executions of the DTW algorithm. Then, we build a 3-dimensional $m \times L \times V$ matrix in which the first entry corresponds to the time series, the second to the attributes and the third to the number V of cells in the warping paths. For each series \mathbf{A}_i we take its value in the entry (i, l, v) of the matrix as the value a_{ilr} , where r is the time period aligned in the v -th cell of the warping path connecting the time series \mathbf{A}_i to the corresponding template $\mathbf{A}_{i'}$. Finally, to obtain a rectangular $m \times n$ matrix, with $n = L \times V$, we proceed by sequencing for each time series \mathbf{A}_i the attributes and the time periods.

3 Temporal discrete support vector machines

At the end of the rectangularization phase, described in section 2, we may assume that the set of time series is represented by a $m \times n$ matrix. In this section, we propose a new mathematical programming model which extends the notion of DSVM in order to perform time series classification.

For many binary classification methods the generic hypothesis takes the form $f(\mathbf{x}) = \text{sgn}(g(\mathbf{x}))$, where $g(\mathbf{x}) : \mathbb{R}^n \mapsto \mathbb{R}$ is a properly defined *score function*. If the space \mathcal{H} is based on the set of separating hyperplanes in \mathbb{R}^n , we have $g(\mathbf{x}) = \mathbf{w}'\mathbf{x} - b$ and $f(\mathbf{x}) = \text{sgn}(\mathbf{w}'\mathbf{x} - b)$. In order to choose the optimal parameters \mathbf{w} and b , SVM resort to the minimization of the following risk functional (Vapnik, 1995; Cristianini and Shawe-Taylor, 2000)

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

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 (Berg *et al.*, 1984) and plays a regularization role; $L(y, f(\mathbf{x}))$ 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 theory of SVM, the loss function measures the distance of the misclassified examples from the separating hyperplane, and is given by

$$L(y, f(\mathbf{x})) = \sum_{i \in \mathcal{M}} |1 - y_i g(\mathbf{x}_i)|_+, \quad (5)$$

where g is a score function such that $f(\mathbf{x}) = \text{sgn}(g(\mathbf{x}))$ and $|t|_+ = t$ if t is positive and zero otherwise. According to DSVM, the loss is instead represented by a discrete function which counts the number of misclassified examples, and given by

$$L(y, f(\mathbf{x})) = \sum_{i \in \mathcal{M}} c_i \theta(-y_i g(\mathbf{x}_i)), \quad (6)$$

where $\theta(t) = 1$ if t is positive and zero otherwise, while $c_i, i \in \mathcal{M}$, is a penalty for the misclassification of the example \mathbf{x}_i . This leads to the formulation of a mixed-integer programming problem that corresponds to the minimization of (4) with the loss function in (6), with the inclusion of an additional regularization term, representing the number of attributes which define the separating hyperplane. The minimization of 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.

The problem of determining an optimal separating hyperplane is formulated as follows in the DSVM framework. The number of misclassified points is computed by means of the binary variables

$$p_i = \begin{cases} 0 & \text{if } \mathbf{x}_i \text{ is correctly classified} \\ 1 & \text{if } \mathbf{x}_i \text{ is misclassified} \end{cases}, \quad (7)$$

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

$$q_j = \begin{cases} 0 & \text{if } w_j = 0 \\ 1 & \text{if } w_j \neq 0 \end{cases}. \quad (8)$$

Let $h_j, j \in \mathcal{N}$, be the penalty cost of using attribute j . Let 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 following *discrete support vector machines* model can be formulated

$$\min_{\mathbf{w}, b, \mathbf{p}, \mathbf{u}, \mathbf{q}} \frac{\alpha}{m} \sum_{i=1}^m c_i p_i + \beta \sum_{j=1}^n u_j + \gamma \sum_{j=1}^n h_j q_j \quad (\text{DSVM})$$

$$\text{s. t. } y_i (\mathbf{w}'\mathbf{x}_i - b) \geq 1 - S p_i \quad i \in \mathcal{M} \quad (9)$$

$$u_j \leq R q_j \quad j \in \mathcal{N} \quad (10)$$

$$-u_j \leq w_j \leq u_j \quad j \in \mathcal{N} \quad (11)$$

$$\mathbf{u} \geq \mathbf{0}, \quad \mathbf{p}, \mathbf{q} \text{ binaries,}$$

where the family of bounding variables $u_j, j \in \mathcal{N}$, and the constraints (11) are introduced in order to linearize the norm of f in the risk functional (4).

In order to extend model DSVM to temporal classification, 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. For each example \mathbf{x}_i of the training set \mathcal{S}_m define the binary variable

$$s_i = \begin{cases} 0 & \text{if } \mathbf{w}'\mathbf{x}_i - b \geq 0 \\ 1 & \text{if } \mathbf{w}'\mathbf{x}_i - b < 0 \end{cases}, \quad (12)$$

indicating to which of the halfspaces supported by the hyperplane the example belongs.

For each pair of examples $(\mathbf{x}_i, \mathbf{x}_k)$ in \mathcal{S}_m define also the binary variable r_{ik} taking the value 1 if \mathbf{x}_i and \mathbf{x}_k lay inside the same halfspace and the value 0 otherwise,

$$r_{ik} = \begin{cases} 0 & \text{if } \{s_i = 1 \text{ and } s_k = 0\} \text{ or } \{s_i = 0 \text{ and } s_k = 1\} \\ 1 & \text{if } \{s_i = 1 \text{ and } s_k = 1\} \text{ or } \{s_i = 0 \text{ and } s_k = 0\} \end{cases}. \quad (13)$$

Let d_{ik} be the DTW distance between the pair of time series $(\mathbf{A}_i, \mathbf{A}_k)$ and Q be a sufficiently large constant value. We can now formulate the following optimization problem termed *temporal discrete support vector machines* (TDVM):

$$\min_{\mathbf{w}, b, \mathbf{p}, \mathbf{u}, \mathbf{q}, \mathbf{r}} \frac{\alpha}{m} \sum_{i=1}^m c_i p_i + \beta \sum_{j=1}^n u_j + \gamma \sum_{j=1}^n h_j q_j + \delta \sum_{i=1}^m \sum_{k=1}^m d_{ik} r_{ik} \quad (\text{TDVM})$$

$$\text{s. t. } y_i (\mathbf{w}'\mathbf{x}_i - b) \geq 1 - S p_i \quad i \in \mathcal{M} \quad (14)$$

$$\mathbf{w}'\mathbf{x}_i - b \geq -Q s_i \quad i \in \mathcal{M} \quad (15)$$

$$\mathbf{w}'\mathbf{x}_i - b \leq (1 - s_i) Q - \varepsilon \quad i \in \mathcal{M} \quad (16)$$

$$r_{ik} \leq 1 - s_i + s_k \quad i, k \in \mathcal{M} \quad (17)$$

$$r_{ik} \leq 1 + s_i - s_k \quad i, k \in \mathcal{M} \quad (18)$$

$$r_{ik} \geq -1 + s_i + s_k \quad i, k \in \mathcal{M} \quad (19)$$

$$r_{ik} \geq 1 - s_i - s_k \quad i, k \in \mathcal{M} \quad (20)$$

$$u_j \leq R q_j \quad j \in \mathcal{N} \quad (21)$$

$$-u_j \leq w_j \leq u_j \quad j \in \mathcal{N} \quad (22)$$

$$\mathbf{u} \geq \mathbf{0}, \quad \mathbf{p}, \mathbf{q}, \mathbf{r} \text{ binaries.}$$

The last term in the objective function (TDVM) represents the sum of the DTW distances between all the pairs of time series assigned to the same class. The inclusion of this term is motivated by the fact that time series in the same class usually exhibit resemblance in the temporal profiles. Thus, by its minimization, the model aims at deriving a separating hyperplane that is optimal also with respect to the similarity of the time series. Constraints (15) and (16) determine on which halfspace point \mathbf{x}_i lies with respect to the supporting hyperplane, therefore fixing the value of the binary variables s_i . Here $\varepsilon > 0$ is a small constant required since the lower halfspace supported by the hyperplane is open. In practice its value can be set equal to the zero precision used in the solution algorithm. Finally, constraints (17), (18), (19) and (20) are required to force variables r_{ik} to one if and only if $s_i = s_k$.

4 Computational setup and analysis

The two-phase method proposed has been implemented and tested as follows. The rectangularization heuristic procedure computes the constrained DTW distance using a dynamic programming algorithm whose search space is reduced by imposing additional bounding constraints. The experiments conducted indicate that a suitable value of the constant V should vary in the range $(T_{av}, \min\{T_{max}, 1.5T_{av}\})$, where T_{av} and T_{max} are the average and the maximum length of the time series $\{\mathbf{A}_i\}$, respectively. The template time series $\mathbf{A}_{i'}$ is selected for each class among those whose length is closer to the average length within the same class.

Model (TDVM) is a mixed binary programming problem, for which a suboptimal solution can be efficiently obtained by a slight modification of the technique proposed in previous papers for solving other

DSVM models. The method is based on solving a short sequence of relaxed linear programming problems, that leads at each step to fix a number of binary variables.

To validate the proposed method we have applied it to a number of datasets composed by multivariate time series arising in real marketing applications, for retention, cross-selling and acquisition in the telecommunication, automotive and retail industries. For each dataset, ten-fold cross validation was applied to estimate the accuracy achieved. Different settings were setup in order to evaluate the impact of the modeling parameters, such as the rectangularization constant V and the relative weight factors $\alpha, \beta, \gamma, \delta$ appearing in the formulation of model (TDVM).

To make comparisons we have considered alternative classifiers based on a standard rectangularization technique obtained via averaging the attributes on fixed time intervals. On all datasets considered the proposed method achieved a significant increase in the accuracy, ranging between 3% and 5%.

Although these results are still preliminary, and require further support on publicly available multivariate temporal datasets, they seem to indicate that the proposed method, based on rectangularization via constrained DTW and an extended version of DSVM, has a great potential to perform accurate classification of time series.

References

- Berg C., Christensen J. P. R., Ressel P. (1984). *Harmonic analysis on semigroups : theory of positive definite and related functions*. Springer.
- Cristianini N., Shawe-Taylor J. (2000). *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press.
- Kadous M. W., Sammut C. (2005). Classification of multivariate time series and structured data using constructive induction. *Machine Learning*, 58, 179–216.
- Keogh E., Ratanamahatana C. A. (2004). Exact indexing of dynamic time warping. *Knowledge and Information Systems*, 1–29.
- Nanopoulos A., Alcock R., Manolopoulos Y. (2001). Feature-based classification of time-series data. *International journal of computer research*, 49–61.
- Orsenigo C., Vercellis C. (2003). Multivariate classification trees based on minimum features discrete support vector machines. *IMA Journal of Management Mathematics*, 14, 221–234.
- Orsenigo C., Vercellis C. (2004). Discrete support vector decision trees via tabu-search. *Journal of Computational Statistics and Data Analysis*, 47, 311–322.
- Orsenigo C., Vercellis C. (2006a). Accurately learning from few examples with a polyhedral classifier. *Computational Optimization and Applications*. To appear.
- Orsenigo C., Vercellis C. (2006b). Multicategory classification via discrete support vector machines. *Computational Management Science*. To appear.
- Rodriguez J. J., Alonso C. J. (2004). Interval and dynamic time warping-based decision trees. In: *SAC '04: Proceedings of the 2004 ACM symposium on Applied computing*, 548–552. ACM Press.
- Sakoe H., Chiba C. (1978). Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-26, 43–49.
- Vapnik V. (1995). *The nature of statistical learning theory*. Springer Verlag.
- Vlachos M., Hadjieleftheriou M., Gunopulos D., Keogh E. (2006). Indexing multidimensional time-series. *The VLDB Journal*, 15(1), 1–20.
- Wu Y., Chang E. Y. (2004). Distance-function design and fusion for sequence data. *CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management*, 324–333.
- Xi X., Keogh E., Shelton C., Wei L. (2006). Fast time series classification using numerosity reduction. *Proceedings of the 23rd International Conference on Machine Learning*. To appear.