Available online at www.sciencedirect.com
ScienceDirect
Pattern Recognition Letters 29 (2008) 933–937
Pattern Recognition Letters
www.elsevier.com/locate/patrec

A large margin approach for writer independent online handwriting classification

Karthik Kumara *, Rahul Agrawal, Chiranjib Bhattacharyya

Department of Computer Science and Automation, IISc, Bangalore, India

Received 16 January 2007; received in revised form 16 January 2008
Available online 1 February 2008

Communicated by P. Sarkar

Abstract

This paper proposes a new approach for classifying multivariate time-series with applications to the problem of writer independent online handwritten character recognition. Each time-series is approximated by a sum of piecewise polynomials in a suitably defined Reproducing Kernel Hilbert Space (RKHS). Using the associated kernel function a large margin classification formulation is proposed which can discriminate between two such functions belonging to the RKHS. The associated problem turns out to be an instance of convex quadratic programming. The resultant classification scheme applies to many time-series discrimination tasks and shows encouraging results when applied to online handwriting recognition tasks.

Keywords: Time-series classification; Kernels; Online handwriting recognition

1. Introduction

Understanding temporal sequences is an important task having many applications in diverse domains like online handwriting recognition, speech recognition, etc. The task of temporal sequence understanding often boils down to assigning proper labels to such sequences. Characters written on a tablet PC or a handheld PDA by a stylus, is represented by the coordinates of the stylus at different time instants, and can be interpreted as a two dimensional time-series. The online handwriting recognition task is to infer the character given in such a sequence. A similar situation arises in speech processing where utterances of various words, vowels, etc. are represented by multidimensional time-series of LPC coefficients. In general a time-series is represented as a collection of time ordered vectors $\mathscr{F} = \{\mathscr{F}(t_1), \mathscr{F}(t_2), \ldots, \mathscr{F}(t_n)\}$ where $\mathscr{F}$ is the time-series, $\mathscr{F}(t_i)$ denotes a vector sampled at the time instant $t_i \in \mathbb{R}$, $t_i < t_j$ whenever $i < j$ and n is the number of sampled points in the time-series. In this paper, we study the problem of classifying such multivariate time-series and develop algorithms suitable for application to online handwriting recognition on a handheld device like PDA.

The problem of time-series classification has been well studied in the area of speech processing where Hidden Markov Models (HMMs) have emerged as a powerful tool. The applicability of HMMs to online handwriting recognition was explored in (Binsztok and Artières, 2004) with mixed results. In recent times Support Vector Machines (SVMs) (Vapnik, 2000) have emerged as a powerful tool for classifying fixed length vectors. The main research issue in extending the SVM formulation to time-series data is the design of kernel function. The Fisher kernel (Jaakkola and Haussler, 1998) was an important breakthrough, which was successfully applied to classifying protein sequences. Each kernel computation requires two passes of a forward backward algorithm on a pre-trained HMM and hence is computationally heavy. Alternatively it is possible to derive

* Corresponding author. Tel.: +91 9845983390; fax: +91 8022932229.
E-mail address: karthikk@csa.iisc.ernet.in (K. Kumara).
URL: http://mllab.csa.iisc.ernet.in/karthik/handwriting/ (K. Kumara).

0167-8655/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2008.01.016

kernels based on dynamic programming based alignment for classification of sequences (Watkins) Dynamic Time Warping (DTW) is an instance of such an approach, which was explored by Bahlmann et al. (2002) for online handwriting recognition with encouraging results over HMM based classifiers. A variant of DTW which uses clustering techniques was proposed in (Bahlmann and Burkhardt, 2003). The state of the art for online handwriting recognition seems to be the online SNT algorithm (Ratzlaff, 2003).

The work reported in the paper is closely related to Sivaramakrishnan and Bhattacharyya (2004), where each time-series is approximated by a sum of piecewise linear functions and resampled to get a fixed length vector. The obtained vector was used to train a SVM classifier. The approach is a two step procedure where in the first step the time-series is represented as a fixed length vector by interpolation and sampling and in the next step the vector is used with a standard classifier. The main contribution of the paper is to propose a kernel function for time-series which can be used not only for classification, but other applications like clustering, novelty detection, etc. The other important contribution is to generalize the piecewise linear interpolation scheme used in (Sivaramakrishnan and Bhattacharyya, 2004) to piecewise polynomials in a Reproducing Kernel Hilbert Space (RKHS) setting. This is interesting in its own right as it might open up new applications which involve interpolation. Lastly the paper also gives an $O(n)$ algorithm for computing the interpolation step rather than an $O(n^3)$ algorithm proposed in (Sivaramakrishnan and Bhattacharyya, 2004).

This paper is organized as follows. Section 2 details the contributions of the paper, it starts by describing a scheme for interpolation of a time-series by sum of polynomials in a RKHS. The scheme is used to derive a continuous function describing the time-series. A large margin formulation is later derived to discriminate these interpolated functions. Section 3 describes the empirical evaluation of the proposed formulation with the state of the art on several benchmark datasets. Finally we end with the concluding remarks in Section 4.

2. Time-series classification

In this section we start by briefly describing an interpolation scheme due to Moore Moore (1985) (Chapter 9), which uses piecewise polynomial functions as basis functions. The interpolation scheme is used to represent each time-series as a function and using a large margin approach we show how to discriminate between such functions.

2.1. Kernel interpolation

Let $0 \leqslant s_1 < s_2 \ldots s_{n-1} < s_n \leqslant 1$ be a sequence of points and let $\mathscr{F}_i = \mathscr{F}(s_i)$ be the function $\mathscr{F} : [0,1] \to \mathbb{R}$ evaluated at s_i. The interpolation problem can be viewed as approximating $\mathscr{F}$ given $\mathscr{D} = \{(s_i, \mathscr{F}_i) | 1 \leqslant i \leqslant n\}$.

Denote by $\mathscr{H}^q$ the space of all functions $f : [0,1] \to \mathbb{R}$, whose qth derivatives are in $\mathscr{L}_2(0,1)$. It can be shown that $\mathscr{H}^q$ is a RKHS (Moore, 1985) with the inner product defined as

$$\langle f, g \rangle = \sum_{j=0}^{q-1} \frac{f^{(j)}(0) g^{(j)}(0)}{j!} + \int_0^1 f^{(q)}(t) g^{(q)}(t)\,\mathrm{d}t, \tag{1}$$

where $f \in \mathscr{H}^q, g \in \mathscr{H}^q$ and $f^{(r)}(t)$ denotes the rth derivative. The reproducing kernel for the RKHS is

$$R_s^q(t) = \sum_{j=0}^{q-1} \frac{s^j t^j}{(j!)^2} + \int_0^{\min(s,t)} \frac{(s-u)^{q-1}(t-u)^{q-1}\,\mathrm{d}u}{((q-1)!)^2} \tag{2}$$

As special case of the above functions

$$R_s^q(t) = \begin{cases} 1 + \min(s,t) & q = 1 \\ \left.\begin{array}{ll} 1 + st + \frac{st^2}{2} - \frac{t^3}{6} & \text{if } t < s \\ 1 + st + \frac{s^2 t}{2} - \frac{s^3}{6} & \text{if } t > s \end{array}\right\} & q = 2. \end{cases}$$

Fig. 1 shows the basis functions for $R_{0.9}^q(t)$ with $q = 1$ and $q = 2$.

Consider the function $\overline{\mathscr{F}}$

$$\overline{\mathscr{F}}(s) = \sum_{j=1}^{n} c_j R_{s_j}^q(s), \tag{3}$$

where the basis function $R_{s_j}^q(s)$ are piecewise polynomial. For any $0 \leqslant s \leqslant 1$ and $f \in H^q$, the basis functions satisfies, $\langle f, R_s^q \rangle = f(s)$, which is also referred as the RKHS property. To ensure $\overline{\mathscr{F}}$ best approximates $\mathscr{F}$, one can choose c by requiring that, $\langle (\overline{\mathscr{F}} - \mathscr{F}), R_{s_i}^q \rangle = 0 \; \forall i = 1, 2, \ldots, n$. Using Eq. (3) and the RKHS property, the above set of equations reduces to, $\sum_{i=1}^{n} \langle R_{s_i}^q, R_{s_j}^q \rangle c_i = \mathscr{F}(s_j)$ or $\mathscr{F} = Gc$. This is a set of linear equations in c and can be efficiently solved. It can be shown that such a choice of c is optimal and the resulting $\overline{\mathscr{F}}$ best approximates $\mathscr{F}$ given $\mathscr{D}$.

The use of the kernel as specified in Eq. (2), enables us to easily incorporate higher order interpolants without changing the structure of the problem. The coefficients c_i can be calculated by solving linear equations, i.e. $c = G^{-1}\mathscr{F}$. This

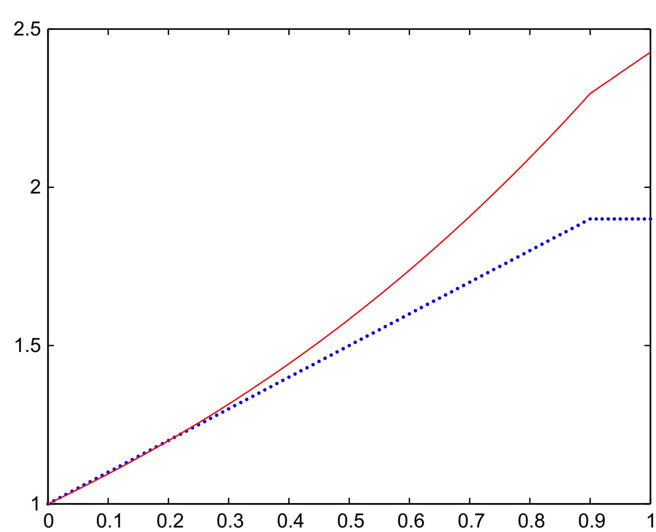

Fig. 1. Plot of basis functions with $q = 1$ and $q = 2$.

can be easily modified to incorporate regularization on the coefficients c as

$$c = \arg\min_{c}(||Gc - \mathscr{F}||^2 + \lambda||c||^2),$$

where λ is the regularization parameter. The ability to incorporate interpolations of different complexities and do regularization are the main advantages of the chosen interpolation kernel.

2.2. Classification of univariate time-series

A univariate time-series $\mathscr{F} = \{\mathscr{F}_1, \ldots, \mathscr{F}_n\}$ $\mathscr{F}_i = \mathscr{F}(t_i)$, evaluated at time instants $\mathscr{F}(t_i) \in \mathbb{R}, 0 \leqslant t_1 < t_2 \cdots < t_n$, can be approximated by a continuous function (3). the basis coefficients c can be chosen to satisfy

$$Gc = [\mathscr{F}_1 \cdots \mathscr{F}_n]^{\mathrm{T}}, \tag{4}$$

where $G_{ij} = \langle R^q_{s_i}, R^q_{s_j}\rangle$, is the Gram Matrix for the basis elements $R^q_{s_i}$. The learning problem can be posed as that of computing a classifier on the dataset $\mathscr{T} = \{(\overline{\mathscr{F}^i}, y_i)|\overline{\mathscr{F}^i} \in \mathscr{H}^q, y_i \in \pm 1\ \forall 1 \leqslant i \leqslant N\}$. More precisely the problem can be stated as finding a $w \in \mathscr{H}^q$ and $b \in \mathbb{R}$ so that the decision function given by, $y = \mathrm{sign}(\langle\overline{\mathscr{F}}, w\rangle + b)$, can correctly predict the class label of a given time-series. A w chosen such that it has the minimum norm $\langle w,w\rangle$ provides the best generalization (Vapnik, 2000). The formulation for finding w and b is

$$\begin{aligned} \min_{w\in\mathscr{H}^q,b} \quad & \frac{1}{2}\langle w, w\rangle \\ \text{subject to} \quad & y_i(\langle w, \overline{\mathscr{F}^i}\rangle + b) \geqslant 1 \\ & 1 \leqslant i \leqslant N \end{aligned} \tag{5}$$

As $w \in \mathscr{H}^q$, one can express w as linear combination of the basis functions in $\mathscr{H}^q$, specifically, $w(t) = \sum_{j=i}^{M} d_j R^q_{m_j}(t)$, where, m_j are normalized time instants such that $m_j \in [0, 1]$. The vector $m = [m_1,\ldots,m_M]$ needs to be chosen appropriately depending on the data. The inner product between two functions $\overline{\mathscr{F}}^1 = \sum_{i=1}^{n1} c_i R^q_{s_i}$ and $\overline{\mathscr{F}}^2 = \sum_{j=1}^{n2} d_j R^q_{s_j}$ is given by

$$\langle\overline{\mathscr{F}}^1, \overline{\mathscr{F}}^2\rangle = \sum_{i=1}^{n_1}\sum_{j=1}^{n_2} c_i d_j \langle R^q_{s_i}, R^q_{s_j}\rangle. \tag{6}$$

Using the above definition the inner product of w and data $\overline{\mathscr{F}}$ is

$$\langle w, \overline{\mathscr{F}}\rangle = \sum_{i=1}^{M}\sum_{j=1}^{n} d_i c_j \langle R^q_{m_i}, R^q_{s_j}\rangle = d^{\mathrm{T}}Ac,$$

where $c = [c_1,\ldots,c_n]$ is a vector determined by (4). The A matrix is given by

$$A_{ij} = \langle R_{m_i}, R_{s_j}\rangle\ 1 \leqslant i \leqslant M, 1 \leqslant j \leqslant n, \tag{7}$$

where M is the number of basis functions describing w. Again from the definition of the inner product (6), the objective can be stated as

$$||w||^2 = \langle w, w\rangle = \sum_{i=1}^{M}\sum_{j=1}^{M} d_i d_j \langle R^q_{m_i}, R^q_{m_j}\rangle. \tag{8}$$

Using (8) the optimization problem (5) can be restated as

$$\begin{aligned} \min_{d,b} \quad & \frac{1}{2}d^{\mathrm{T}}Bd \\ \text{subject to} \quad & y_i(d^{\mathrm{T}}A^{(i)}c^{(i)} + b) \geqslant 1 \\ & 1 \leqslant i \leqslant N, \end{aligned} \tag{9}$$

where B is the Gram matrix for the basis functions given by, $B_{ij} = \langle R^q_{m_i}, R^q_{m_j}\rangle$ from Eq. (8), $A^{(i)}$ and $c^{(i)}$ is the A matrix and c vector, respectively, for the ith data point.

The matrix B is positive semi-definite and hence the optimization problem (9) is an instance of convex quadratic programming and can be solved by standard tools. The decision function can now be evaluated as, $y = \mathrm{sign}(d^{\mathrm{T}}Ac + b)$.

The above formulation (9) provides a unified scheme for time-series classification and is the main contribution of this paper. In the following paragraphs, the above formulation is cast as an optimization problem which can be handled by standard SVM solvers (Chang and Lin, 2001). The formulation is extended to handle multivariate data. A simple $O(n)$ scheme is also suggested for linear interpolation.

2.3. A time-series kernel

The Gram matrix B can be factorized as $B = U\Sigma U^{\mathrm{T}}$. Where U is the matrix formed by normalized eigen vectors and Σ is a diagonal matrix, the eigen values being the diagonal elements. It can be shown that $UU^T = U^{\mathrm{T}}U = I$. Based on this we can extract a kernel from (9) that can be used for comparing similarity between sequences of time-series. Let $d = U^T\Sigma^{-\frac{1}{2}}u$, where $\Sigma^{-\frac{1}{2}}$ is the inverse of the matrix square root of the diagonal matrix Σ. Substituting in (9) one obtains the following formulation:

$$\begin{aligned} \min_{u,b} \quad & \frac{1}{2}u^{\mathrm{T}}u \\ \text{subject to} \quad & y_i(u^{\mathrm{T}}X_i + b) \geqslant 1, \\ & 1 \leqslant i \leqslant N, \end{aligned} \tag{10}$$

where $X_i = \Sigma^{-\frac{1}{2}}UA^{(i)}c^{(i)}$. The matrix $\Sigma^{-\frac{1}{2}}U$ can be precomputed for faster operation. The formulation (10) is equivalent to an SVM (Vapnik, 2000) with the Kernel, $K(X_i, X_j) = X_i^{\mathrm{T}}X_j$.

In the case of handwritten data the PDA gives two time-series, one for each coordinate or in case of speech data there can be multiple time varying features. The theory developed so far can handle only univariate time-series. In the following we generalize the approach to handle multivariate time-series.

In general consider a vector of functions with L dimensions, $\overline{\mathscr{F}} = [\overline{\mathscr{F}}_1, \ldots, \overline{\mathscr{F}}_L]^{\mathrm{T}}$. The inner product of two such functions F and G, each with L dimension $\langle F, G\rangle = \sum_{l=1}^{L}\langle F_l, G_l\rangle$. Using this definition one can compute the kernel as

$$K(i,j) = \sum_{l=1}^{L} X_{i_l}^{\mathrm{T}} X_{j_l} = X_i^{\mathrm{T}} X_j, \qquad (11)$$

where $X_i = [X_{i_1}^{\mathrm{T}}, \ldots, X_{i_l}^{\mathrm{T}}]^{\mathrm{T}}$, which is equivalent to concatenating the univariate vectors X_{i_l}.

As in standard SVM procedure one can derive the dual of the formulation in Eq. (10) and relax the formulation to handle non-separable data by using any positive definite kernel and restricting the α_is to be less than a user defined constant $C > 0$ (Vapnik, 2000)

$$\begin{aligned} \min_{\alpha} \quad & \frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j y_i y_j K(X_i, X_j) - \sum_i \alpha_i \\ \text{subject to} \quad & \sum_i y_i \alpha_i = 0 \\ & 0 \leqslant \alpha_i \leqslant C, 1 \leqslant i \leqslant N. \end{aligned} \qquad (12)$$

We have experimented with the Radial Basis kernel, $K(X_i, X_j) = \mathrm{e}^{-\gamma \|X_i - X_j\|^2}$.

Finally we are ready to state our classification algorithm.

Algorithm 1. Algorithm for classification of time-series.

Given: Training Data $\mathscr{T}$. Parameters: Sampling Instants m, Kernel Order q.
Re-scale time axis to $[0, 1]$ for each time-series $\mathscr{F}^i$.
Find the coefficients $c^{(i)}$ by solving (4).
Find $A^{(i)}$ for the ith time-series at time instants specified by m (see (7)).
Calculate X_i using $X_i = \Sigma^{-\frac{1}{2}} U A^{(i)} c^{(i)}$.
Use the appropriate kernel (11) or (2.3) and solve using SVM.

2.4. Fast inversion scheme for piecewise linear basis functions

The calculation of vectors X_i, depends on time-series interpolation, which is a costly task. However, for piecewise linear basis functions, $q = 1$ (see Eq. (1)), it is possible to do the interpolation cheaply. Note that the basis functions can be written as

$$R_t(s) = \begin{cases} 1 + s & 0 \leqslant s \leqslant t \\ 1 + t & t \leqslant s \leqslant 1. \end{cases}$$

The structure of the basis functions can be exploited to obtain a recursive formula for c_i which leads to a fast algorithm for interpolation.

Consider a function f whose values are available at various time instants, s_i, $1 \leqslant i \leqslant n$. We are interested in finding c_i such that $f(s) = \sum_i c_i R_{s_i}(s)$. See that $s_i < s_j$ whenever $i < j$ which gives $f(s_j) = \sum_{i<j} c_i (1 + s_i) + (1 + s_j)(\sum_{i \geqslant j} c_i)$. Also note that, $f(s_{j-1}) = \sum_{i<j-1} c_i (1 + s_i) + (1 + s_{j-1})(\sum_{i \geqslant j-1} c_i)$. Subtracting one from the other we have, $f(s_j) - f(s_{j-1}) = (s_j - s_{j-1}) \sum_{i \geqslant j} c_i$. The coefficients c can be calculated as follows:

$$c_j = \frac{f(s_j) - f(s_{j-1})}{s_j - s_{j-1}} - \sum_{i>j} c_i. \qquad (13)$$

The boundary conditions being, $c_n = \frac{f(s_n) - f(s_{n-1})}{s_n - s_{n-1}}$ and $c_1 = \frac{f(s_1)}{1+s_1} - \sum_{i>1} c_i$. This sets up a recursive formula for calculating c_i, see that it proceeds backwards starting from n. The worst case time complexity of the above algorithm is $O(n)$ which is again considerably cheaper than the matrix inversion step.

2.5. A comparison with resampled approach

In (Sivaramakrishnan and Bhattacharyya, 2004) a two step scheme for interpolation and resampling was used with SVM classification. The classification kernel in such a case is a simple L2 product of the resampled vector. $K_{ij} = (A^{(i)} c^{(i)})^T (A^{(j)} c^{(j)})$, where c is the coefficients of basis functions (4), and A is as defined in (7). In the proposed formulation, the kernel obtained after optimization is given by, $K_{ij} = X_i^T X_j = (A^{(i)} c^{(i)})^T B^{-1} (A^{(j)} c^{(j)})$. Thus, if the matrix B is identity, then the two formulations are identical.

In this section, we have proposed a formulation for classifying time-series and discussed several ways in which the formulation can be practically applied to real world problems. One of the strengths of our formulation is that we can employ polynomials of any positive order. It must be noted that as the order increases, the piecewise polynomials attain increasingly complicated shapes. Trade off must be decided between the complexity of the piecewise polynomial and the extent of fitting that is done.

3. Experiments

In this section, the efficacy of the large margin classifier proposed in the previous section is tested on three benchmark datasets from Unipen (Guyon et al., 1994) and the results are compared with other state of the art algorithms such as CSDTW (Bahlmann and Burkhardt, 2003) and Online SNT algorithm (Ratzlaff, 2003).

Experiments were conducted on the standard train_r01_v07 package from unipen, more specifically experiments were conducted on three datasets namely, 1a (digits), 1b (uppercase alphabets) and 1c (lowercase alphabets). Results from other authors (Ratzlaff, 2003) are available on these datasets for comparison.

The data needs to be preprocessed before applying the proposed formulation (12). The first step involves converting the unipen form data into a time-series containing a series of $[x, y, t]$ for each character. This is achieved by extracting strokes from the data, and obtaining the time-series for each of the strokes. Time-series for a character is obtained by concatenating the time-series obtained by individual strokes in the character. Each of the series is further normalized in x, y, such that the character lies in a fixed size box of 50×50 points. The scaling is done such that the aspect ratio of the character was maintained.

Table 1
QP formulation with a Gaussian kernel with first and second order interpolation compared with other state of the art methods

Data	Approach	Error (%)	$q=1$	$q=2$
Unipen dataset (R01/V07) 1a	QP Gaussian	2.87	2.87	3.9
Digits (0–9)	CSDTW (Bahlmann and Burkhardt, 2003)	2.9		
	DAG–SVM–GDTW (Bahlmann et al., 2002)	3.8		
	HMM (Hu et al., 2000)	3.2		
	OnSNT (Ratzlaff, 2003)	1.1		
Unipen dataset (R01/V07) 1b	QP Gaussian	8.52	8.52	11.46
Upper case alphabets (A–Z)	CSDTW (Bahlmann and Burkhardt, 2003)	7.2		
	DAG–SVM–GDTW (Bahlmann et al., 2002)	7.6		
	HMM (Hu et al., 2000)	6.4		
	OnSNT (Ratzlaff, 2003)	4.5		
Unipen dataset (R01/V07) 1c	QP Gaussian	10.78	10.78	13.76
Lower case alphabets (a–z)	CSDTW (Bahlmann and Burkhardt, 2003)	9.3		
	DAG–SVM–GDTW (Bahlmann et al., 2002)	12.1		
	HMM (Hu et al., 2000)	14.1		
	OnSNT (Ratzlaff, 2003)	7.9		

The character is also translated on both x and y axes such that the minimum value for these axes is zero. The code for preprocessing, training and testing are available for download from our site (Code for QP formulation).

The preprocessed data was randomly divided into three sets, with 50% of the data used for training, 16% for validation and the remaining 33% of the data used as the unseen test data set. Classifiers were tuned on the validation set to obtain the best values for the classifier parameters γ of the Gaussian kernel, m the number of basis vectors in the discriminating function and c the cost parameter of SVM. The classifiers so tuned was used for classification of the unseen test data. The results of the experiments along with results from previous work are provided in Table 1.

In the previous section it was shown that (9) can be solved by standard SVM solvers. The experiments were carried out by adding these custom kernels to the libsvm package (Chang and Lin, 2001). The kernel in Eq. (11) takes different forms with different order of interpolation. We have limited our experiments up to second order interpolants ($q=1$, $q=2$ in Eq. (1)). The experiments can be easily extended to higher order interpolants.

The performance of our QP formulation is competitive to state of the art methods in all the three benchmark datasets. The performance of the linear interpolation is slightly better than that of the quadratic interpolation scheme. This may be specific to the online handwriting domain.

4. Conclusion

In this paper a large margin based classification scheme is described for classifying multivariate time-series. The scheme proceeds by representing each time-series as a sum of piecewise polynomial basis functions through a kernel interpolation technique. Using the kernel a large margin formulation is developed which is capable of classifying such interpolated functions. The formulation is shown to be competitive in writer independent handwriting recognition on four real world datasets. Initial experiments have shown that the scheme is profitable in any other domain like speaker identification. A set of kernels were also extracted for classification with standard tools, the kernels can also be used in other machine learning tasks. We also give a fast algorithm for computing the interpolation when piecewise linear basis functions are used. As a by-product the kernel interpolation described can be useful in many other machine learning tasks.

References

Bahlmann, C., Burkhardt, H., 2003. The writer independent online handwriting recognition system frog on hand and cluster generative statistical dynamic time warping. IEEE Trans. Pattern Anal. Machine Intell. 26 (3), 299–310.

Bahlmann, C., Haasdonk, B., Burkhardt, H., 2002. On-line handwriting recognition with support vector machines: A kernel approach. In: Internat. Workshop on Frontiers in Handwriting Recognition, pp. 49–54.

Binsztok, H., Artières, Thierry, 2004. Learning HMM structure for online handwriting modelization. In: Internat. Workshop on Frontiers in Handwriting Recognition, Tokyo, pp. 407–412.

Chang, C.-C., Lin, C.-J., 2001. LIBSVM: a library for support vector machines, software available at <http://www.csie.ntu.edu.tw/cjlin/libsvm>.

Code for QP formulation available at: <http://mllab.csa.iisc.ernet.in/users/karthik/handwriting/>.

Guyon, I., Schomaker, L., Plamondon, R., Liberman, M., Janet, S., 1994. Unipen project of on-line data exchange and recognizer benchmarks. In: Proc. Internat. Conf. on Pattern Recognition, pp. 29–33.

Hu, J., Lim, S.G., Brown, M.K., 2000. Writer independent on-line handwriting recognition using an HMM approach. Pattern Recognition 33 (1), 133–147.

Jaakkola, T., Haussler, D., 1998. Exploiting generative models in discriminative classifiers. In: Adv. Neural Inform. Process. Syst., vol. 11. MIT Press.

Moore, R.E., 1985. Computational Functional Analysis. Academic Press, NY.

Ratzlaff, E.H., 2003. Methods, report and survey for the comparison of diverse isolated character recognition results on the UNIPEN database. In: Internat. Conf. on Document Analysis and Recognition, pp.623–628.

Sivaramakrishnan, K.R., Bhattacharyya, C., 2004. Time series classification for online tamil handwritten character recognition – A kernel based approach. In: ICONIP, pp. 800–805.

Vapnik, N.V., 2000. The Nature of Statistical Learning Theory. Springer-Verlag, New York.

Watkins, C., Dynamic alignment kernels, Tech. Rep. CSD-TR-98-11, Department of Computer Science, University of London, Egham, England