Representing Functional Data Using Support Vector Machines

Javier González and Alberto Muñoz

Universidad Carlos III de Madrid, c/ Madrid 126, 28903 Getafe, Spain
{javier.gonzalez,alberto.munoz}@uc3m.es

Abstract. Functional data are difficult to manage for many traditional pattern recognition techniques, given the very high (or intrinsically infinite) dimensionality. The reason is that functional data are functions and most algorithms are designed to work with (small) finite-dimensional vectors. In this paper we propose a functional analysis technique to obtain finite-dimensional representations of functional data. The key idea is to consider each functional curve as a point in a general function space and then project these points onto a Reproducing Kernel Hilbert Space with the aid of a Support Vector Machine. We show some theoretical properties of the method and illustrate the performance of the proposed representation in clustering using a real data set.

Keywords: Support Vector Machines, Reproducing Kernel Hilbert Spaces, Functional Data.

1 Introduction

The field of Functional Data Analysis (FDA) [7] deals naturally with data of very high (or intrinsically infinite) dimensionality. Typical examples are functions describing processes of interest, such as physical processes, genetic data, control quality charts or spectral data in chemometrics. In practice a functional datum comes as a set of discrete measured values. FDA methods first convert these values to a function and then apply some generalized multivariate procedure able to cope with functions.

The key idea in our proposal is to see each function as a point in a given function space, and then to project these points onto some finite-dimensional function subspace. The best known example of such spaces are Reproducing Kernel Hilbert Spaces (RKHS), and the usual approach to represent functions as points in RKHS is Regularization Theory. Details will be given in Section 2. In particular, we propose a finite-dimensional representation for functional data based on a particular projection (given by the use of a Support Vector Machine) of the original functions onto the subspace generated by the eigenfunctions of the RKHS kernel.

In Section 2 we formulate the functional data representation in the context of regularization theory, for the ϵ-insensitive loss function (the SVM loss function). We also show how to approximate the eigenfunctions of the kernel when working

J. Ruiz-Shulcloper and W.G. Kropatsch (Eds.): CIARP 2008, LNCS 5197, pp. 332–339, 2008.

with finite samples and/or sample kernel matrices. In Section 3 we illustrate the performance of the proposed functional data representation in a clustering task using a data set of temperature series from weather stations in Canada. Section 4 outlines some future research lines and concludes.

2 Representing Functional Data in a Reproducing Kernel Hilbert Space

We want to transform each curve (functional datum) into a point of a RKHS. Let $\{\hat{c}_1, \ldots, \hat{c}_m\}$ denote the available sample of curves. Each sampled curve $\hat{c}_l$ is identified with a data set $\{(\mathbf{x}_i, \mathbf{y}_{il}) \in X \times Y\}_{i=1}^n$. X is the space of input variables and, in most cases, $Y = \mathbb{R}$. We assume that, for each $\hat{c}_l$, there exists a continuous function $c_l : X \longrightarrow Y$ such that $E[y_l|\mathbf{x}] = c_l(\mathbf{x})$ (with respect to some probability measure). Thus $\hat{c}_l$ is the sample version of c_l. Notice that, for simplicity in notation, we assume that the $\mathbf{x}_i$ are common for all the curves, as it is the habitual case in the literature [7].

There are several ways to introduce RKHS (see [1,3,9,5]). In a nutshell, the essential ingredient for a Hilbert function space H to be a RKHS is the existence of a symmetric positive definite function $K : X \times X \to \mathbb{R}$ named Mercer Kernel or reproducing kernel for H [1]. The elements of H, H_K in the sequel, can be expressed as finite linear combinations of the form $h = \sum_s \lambda_s K(x_s, \cdot)$ where $\lambda_s \in \mathbb{R}$ and $x_s \in X$.

Consider the linear integral operator T_K associated to K defined by $T_K(f) = \int_X K(\cdot, s) f(s) ds$. If we impose that $\int\int K^2(x, y) dx dy < \infty$, then T_K has a countable sequence of eigenvalues $\{\lambda_j\}$ and (orthonormal) eigenfunctions $\{\phi_j\}$ and K can be expressed as $K(x, y) = \sum_j \lambda_j \phi_j(x) \phi_j(y)$ (where the convergence is absolute and uniform).

Given a function f in a function space containing H_K, it will be projected to H_K using the operator T_K. Thus, the projection f^* will belong to the range of T_K: $f^* = T_K(f)$. Applying the Spectral Theorem to T_K we get:

$$f^* = T_K(f) = \sum_j \lambda_j \langle f, \phi_j \rangle \phi_j \tag{1}$$

Next we want to obtain c_l^* for each c_l (the function corresponding to the sample functional data point $\hat{c}_l \equiv \{(\mathbf{x}_i, y_{il}) \in X \times Y\}_{i=1}^n$). To find the coefficients of c_l^* in eq. (1), we apply a SVM to express the approximation of $\hat{c}_l$ in terms of a kernel expansion. To this aim, the SVM seeks the function c_l^* that solves the following functional optimization problem [3,5] :

$$\arg \min_{c \in H_K} \frac{1}{n} \sum_{i=1}^{n} L(y_i, c(\mathbf{x}_i)) + \gamma \|c\|_K^2 . \tag{2}$$

where $\gamma > 0$, $\|c\|_K$ is the norm of the function c in H_K, $y_i = \hat{c}_i$ and $L(y_i, c(\mathbf{x}_i)) = (|c(\mathbf{x}_i) - y_i| - \varepsilon)_+$, $\varepsilon \geq 0$ in the SVM approach [5]. Expression (2) measures the

trade-off between the fitness of the function to the data and the complexity of the solution (measured by $\|c\|_K^2$). By the Representer Theorem ([4,8]), the solution c_l^* to the functional optimization problem (2) exists, is unique and admits a representation of the form

$$c_l^*(\mathbf{x}) = \sum_{i=1}^{n} \alpha_{il} K(\mathbf{x}_i, \mathbf{x}), \quad \forall \mathbf{x} \in X \text{ where } \alpha_i \in \mathbb{R} \,. \tag{3}$$

Next we define two functional data representations starting from eq. (3).

2.1 Functional Data Projections in the Eigenfunctions Space

By minimizing the risk functional (2) we obtain the points $c_1^*, \ldots c_m^*$ in H_K corresponding to the original curves $\{\hat{c}_1, \ldots, \hat{c}_m\}$. Equation (3) gives a first approximation to the representation of each curve $\hat{c}_l$, namely the set of coefficients $\alpha_{1l}, \ldots, \alpha_{nl}$. However, this representation has a serious drawback: it is not a continuous function in the input variables and, therefore, if we have a slightly different sample $(\mathbf{x}_i')$, may be that the corresponding y_{il}' are quite different, making the representation system not valid for pattern recognition purposes:

Theorem 1. *Let c be a curve, whose sample version is $\{(\boldsymbol{x}_i, y_i) \in X \times Y\}_{i=1}^n$. The representation of c in terms of the vector $(\alpha_1, \ldots, \alpha_n)$ is not continuous, where $\{\alpha_i\}_{i=1}^n$ are the coefficients of c^* in (3) solved using the Support Vector Machine: $c^*(\boldsymbol{x}) = \sum_{i=1}^n \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x})$.*

Proof. The number of non null terms in the sum $K(\mathbf{x}, \mathbf{y}) = \sum_i \lambda_i \phi^T(\mathbf{x})\phi(\mathbf{y})$ is $d = \min(n, r(T_K))$, where $r(T_K)$ is the rank of the operator T_K ($\lambda_j = 0$ for $j > r(T_K)$).

$$\begin{aligned} c^*(\mathbf{x}) &= \sum_{i=1}^{n} \alpha_i k(\mathbf{x}_i, \mathbf{x}) = \sum_{i=1}^{n} \alpha_i \left(\sum_{j=1}^{d} \lambda_j \phi_j(\mathbf{x}_i) \phi_j(\mathbf{x}) \right) \\ &= \sum_{j=1}^{d} \lambda_j \left(\sum_{i=1}^{n} \alpha_i \phi_j(\mathbf{x}_i) \right) \phi_j(\mathbf{x}) \end{aligned} \tag{4}$$

On the other hand, by equation (1) $c^*(\mathbf{x}) = \sum_{j=1}^d \lambda_j \langle c, \phi_j \rangle \phi_j(\mathbf{x})$. Equating to (4) and being the $\{\phi_j\}$ a basis for H_K we get: $\langle g, \phi_j \rangle = \sum_i \alpha_i, \phi_j(\mathbf{x}_i) = \langle \alpha, \phi_j \rangle$. Therefore, for any set $\alpha' = (\alpha_1', \ldots, \alpha_n')$ such that $\langle \alpha', \phi_j \rangle = \langle \alpha, \phi_j \rangle = \langle g, \phi_j \rangle$ we will have that $\sum_{i=1}^n \alpha_i' k(\mathbf{x}_i, \mathbf{x}) = c^*(\mathbf{x})$. Now, given the sample curve $c \equiv \{(\mathbf{x}_i, y_i) \in X \times Y\}_{i=1}^n$, consider a 'close' curve $c^\epsilon \equiv \{(\mathbf{x}_i^\epsilon, y_i^\epsilon) \in X \times Y\}_{i=1}^n$, such that $d(\mathbf{x}, \mathbf{x}^\epsilon) < \epsilon$. Denote by (α^ϵ) the representation corresponding to c^ϵ. Given that $c^\epsilon(\mathbf{x}) \simeq c^*(\mathbf{x})$ (because of the continuity of c), and using eq. (4) it will happen that $\langle \alpha^\epsilon, \phi_j \rangle \simeq \langle \alpha, \phi_j \rangle$ and, nevertheless, by the previous reasoning, α^ϵ and α can be quite different. From an intuitive point of view, think that if there is a small change in the sample, then one or more support vectors can change and, therefore, the α_i (that are associated to the support vectors) will change.

The next theorem specifies our concrete proposal to obtain functional data representations.

Theorem 2. *Let c be a curve, whose sample version is $\hat{c} \equiv \{(\boldsymbol{x}_i, y_i) \in X \times Y\}_{i=1}^n$. Consider the functional representation for c given by $(\lambda_1^*, \ldots, \lambda_d^*)$, where*

$$\lambda_j^* = \sum_{i=1}^{n} \hat{\lambda}_j \alpha_i \hat{\phi}_{ji} \ , \tag{5}$$

α_i are given by (3), $\hat{\lambda}_j$ is the eigenvalue corresponding to the eigenvector $\hat{\phi}_j$ of the matrix $K_S = (K(\boldsymbol{x}_i, \boldsymbol{x}_j))_{i,j}$, and $d = \min(n, r(K_S))$. This functional representation is continuous in the input variables.

Proof. In the ideal case where we know the expression for both the eigenfunctions and eigenvalues of the kernel function K, $\lambda_j^* = \sum_{i=1}^n \lambda_j \alpha_i \phi_j(\mathbf{x}_i)$. However, often we only know the matrix K_S, obtained by evaluating the kernel at the sample, and we can not know the real eigenvalues λ_j and their corresponding eigenfunctions ϕ_j. We will prove the theorem for the representation given by $\sum_{i=1}^n \lambda_j \alpha_i \phi_j(\mathbf{x}_i)$, and then we show that $\sum_{i=1}^n \hat{\lambda}_j \alpha_i \hat{\phi}_{ji}$ converges to that value. First we show that $\sum_{j=1}^d \lambda_j^* \phi_j(\mathbf{x})$ gives the value of $c^*(\mathbf{x})$:

$$\begin{aligned} \sum_{j=1}^{d} \lambda_j^* \phi_j(\mathbf{x}) &= \sum_{j=1}^{d} \left(\lambda_j \sum_{i=1}^{n} \alpha_i \phi_j(\mathbf{x}_i) \right) \phi_j(\mathbf{x}) = \sum_{j=1}^{d} \lambda_j \left(\sum_{i=1}^{n} \alpha_i \phi_j(\mathbf{x}_i) \right) \phi_j(\mathbf{x}) \\ &= \sum_{i=1}^{n} \alpha_i \left(\sum_{j=1}^{d} \lambda_j \phi_j(\mathbf{x}_i) \phi_j(\mathbf{x}) \right) \sum_{i=1}^{n} \alpha_i k(\mathbf{x}_i, \mathbf{x}) = c^*(\mathbf{x}) \end{aligned} \tag{6}$$

Given the sample curve c and a 'close' curve c^ϵ, and using the same notation as in Theorem 1, if $d(x, x^\epsilon) < \epsilon$ then $c(\mathbf{x}) \simeq c^\epsilon(\mathbf{x})$ and given that the ϕ_j are a basis for H_K, it must happen that $\lambda_j^* \simeq \lambda_j^{*\epsilon}$. To end the proof, we only need to show that the eigenvalues and eigenvectors of K_S converge, respectively, to the eigenvalues and eigenfunctions of T_K: $\hat{\lambda}_j \longrightarrow \lambda_j$ and $\hat{\phi} \longrightarrow \phi$. And this is the case because this convergence holds always for positive-definite matrices, including kernel functions (see [6]). For more specific theorems restricted to the context of kernel functions, see [2].

2.2 Truncation Error Analysis

Given that the kernel expansion $K(\mathbf{x}, \mathbf{y}) = \sum_{j=1} \lambda_j \phi_j(\mathbf{x}) \phi_j(\mathbf{y})$ contains at most $d = \min(n, r(K_S))$ non null terms, we wonder about the quality of the approximation of $K^{[d]}(\mathbf{x}, \mathbf{y}) = \sum_{j=1}^d \lambda_j \phi_j(\mathbf{x}) \phi_j(\mathbf{x})$ to $K(\mathbf{x}, \mathbf{y})$. If $d = r(K)$, then $K^{[d]} = K$ and there is no loss in using $K^{[d]}$. If $d = n$, then the number of eigenfunctions is larger than the number of data points and $K^{[d]}$ takes only into

account the first n eigenvalues of K, and calling $c^{*[n]}(\mathbf{x}) = \sum_{j=1}^{d} \lambda_j^* \phi_j(\mathbf{x})$, is immediate to prove that the truncation error is given by

$$E_r = \|c^* - c^{*[r]}\|^2 = \sum_{j=d+1} \lambda_j^{*2} \ . \tag{7}$$

As n the number of data points increases, $\lambda_j \to 0$ (because the sum that defines K converges) and so does the truncation error.

3 Experiments

In this first example we illustrate the behaviour of the functional data representation with a simple data set. Figure 1, left, shows five logistic functions, identical except for a shift, sampled at 200 points. Figure 1, right, shows the functional representation for the five logistic functions, using a exponential kernel. There are eleven non zero eigenfunctions, but there are only three active dimensions for the this data set. The vectors representing the five functions are all very similar, except in the seventh dimension, that seems to have captured the only difference among the functions. Thus, in the functional data space, the functions are represented by very similar vectors, except in one component.

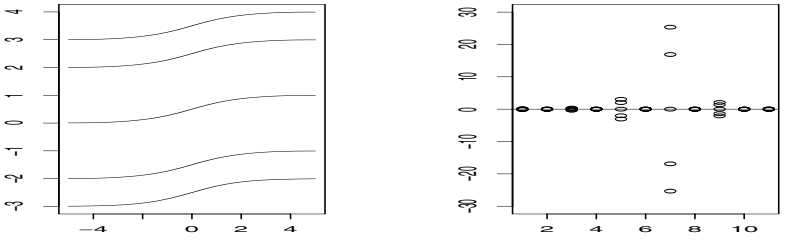

Fig. 1. Several logistic functions and their functional representations

3.1 Canadian Temperature Data Set

In this example we focus on a data set corresponding to daily series (thus sampled at 365 points) of temperature taken at 35 different locations in Canada averaged over the period from 1960 to 1994. This data set has been analyzed in the past [7], but its structure has not been explained up to date. Next we show that the functional representation we propose is the key to reveal the structure in this functional data set.

We project the temperature curves to the RKHS defined by $K(x, y) = e^{\{-\gamma \|x-y\|^2\}}$, with $\gamma = 0.01$ and calculate the coefficients α_{il} for the curves using a Regression SVM with $\epsilon = 1$ and $C = 10$ (parameters fixed by cross validation). The whole set of curves and the point in the RKHS corresponding to the city of Quebec are shown in Figure 2. Three groups can be considered in term of the

geographical locations of the stations. The first group is formed by 21 locations in the south of the country, that do not suffer extremely low temperatures. The next group, made up of Vancouver, Victoria and Pr. Rupert, contains cities with a mild climate, and they located at the southwestern. The remaining group corresponds to 11 locations in the north of the country and exhibit more rigorous winters. Figure 5 shows the location of the temperature stations.

The functional representation using coefficients in eq. (5) produces 175 nonzero eigenfunctions but only the first 50 components point out differences between the curves. Using the 175-dimensional representations for the temperature curves, a simple hierarchical cluster method (Ward), is able to recover the three described groups (with the exception of the cities of Winnipeg and Regina). The curves within each cluster are shown in Figure 3. Notice that the range of temperatures exhibited by curves within each obtained cluster fully agrees with the described groups in the previous paragraph.

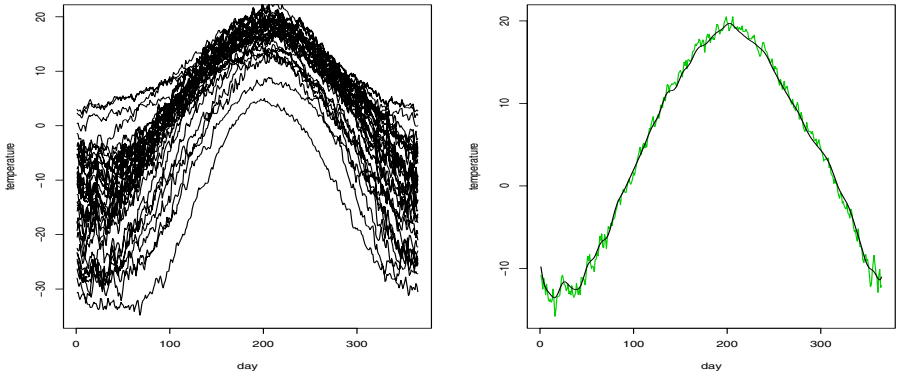

Fig. 2. Temperature data at 35 Canada stations (left) and the serie and its regularized version for the city of Quebec

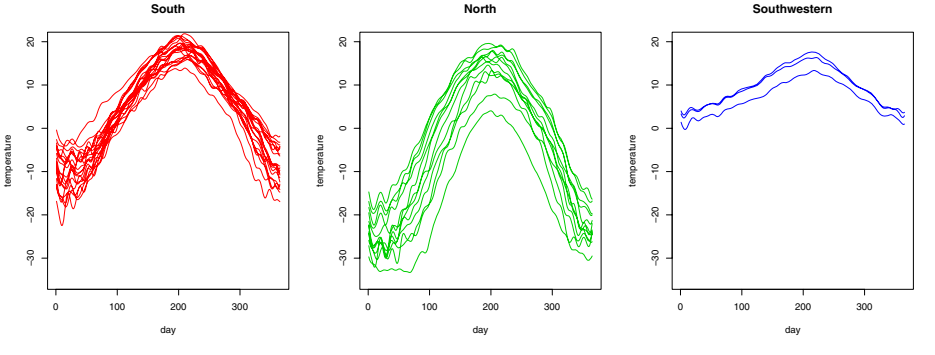

Fig. 3. Three clusters of temperature curves using the functional representation given by eq. (5)

If we use the representation given by the α_i in eq. (3) the cluster analysis fails to recover the prespecified clusters, as can be concluded by looking at the temperature ranges of the curves in the groups shown in Figure 4. The success of each representation method is summarized in Table 1.

Table 1. Number of errors in the cluster analysis for the three representation systems. The 'true' labels are that shown in Figure 5.

Functional Data Representation	**Raw Data**	**Kernel expansion** (α_i)	**RKHS representation**(λ_i^*)
Classification Errors	12	21	2

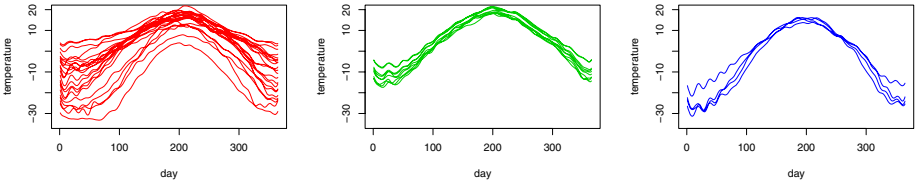

Fig. 4. Three clusters of the temperature curves using the coefficients of the kernel expansion in eq. (3)

Fig. 5. Map of Canada with the three a priori clusters

4 Conclusions

In this work we have proposed a system to represent functional data, by projecting the original functions onto the eigenfunctions of a Mercel kernel. The projection is achieved by using Support Vector Machines. A main advantage is that we do not have to specify the basis of eigenfunctions, but we can concentrate in the kernel, following the general philosophy of kernel methods. The proposed representation seems to work well in the experiments, capturing the interesting features of functional data and performing well in clustering tasks.

Regarding future work, we want to investigate the choice of kernels appropriate for preespecified tasks or data sets. The idea is to specify objective functions in terms of distance criteria (as it happens, for instance, for principal component analysis). Given the direct relationship existing between kernel functions and distance functions, this gives as a method to specify optimal kernels in advance and to obtain, in consequence, optimal representation systems for given tasks.

Acknowledgments

This work was partially supported by Spanish grant SEG 2007/64500.

References

1. Aroszajn, N.: Theory of Reproducing Kernels. Transactions of the American Mathematical Society 68(3), 337–404 (1950)
2. Bengio, Y., Delalleau, O., Le Roux, N., Paiement, J.-F., Vincent, P., Ouimet, M.: Learning eigenfunctions links spectral embedding and kernel PCA. Neural Computation 16, 2197–2219 (2004)
3. Cucker, F., Smale, S.: On the Mathematical Foundations of Learning. Bulletin of the American Mathematical Society 39(1), 1–49 (2002)
4. Kimeldorf, G.S., Wahba, G.: A Correspondence between Bayesian Estimation on Stochastic Processes and Smoothing by Splines. Annals of Mathematical Statistics 2, 495–502 (1971)
5. Moguerza, J.M., Muñoz, A.: Support Vector Machines with Applications. Statistical Science 21(3), 322–357 (2006)
6. Schlesinger, S.: Approximating Eigenvalues and Eigenfunctions of Symmetric Kernels. Journal of the Society for Industrial and Applied Mathematics 6(1), 1–14 (1957)
7. Ramsay, J.O., Silverman, B.W.: Functional Data Analysis, 2nd edn. Springer, New York (2006)
8. Schölkopf, B., Herbrich, R., Smola, A.J., Williamson, R.C.: A Generalized Representer Theorem. In: Helmbold, D.P., Williamson, B. (eds.) COLT 2001 and EuroCOLT 2001. LNCS (LNAI), vol. 2111, pp. 416–426. Springer, Heidelberg (2001)
9. Wahba, G.: Spline Models for Observational Data. Series in Applied Mathematics, vol. 59. SIAM, Philadelphia (1990)