Combining Functional Data Projections for Time Series Classification

Alberto Muñoz and Javier González

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

Abstract. We afford the classification of time series in the Functional Data Analysis (FDA) context. To this aim we introduce projections methods for the time series onto appropriate Reproducing Kernel Hilbert Spaces (RKHSs) with the aid of Regularization Theory. Next we project the curves onto a set of different RKHSs. Then we consider the induced Euclidean metrics in these spaces and combine them in order to obtain a single kernel valid for classification purposes. The methodology is tested on some real and simulated classification examples.

Keywords: Functional data, Regularization Theory, Reproducing Kernel Hilbert Spaces, Kernel Combination, Classifier Fusion.

1 Introduction

The field of Functional Data Analysis (FDA) [12,6] deals naturally with data of very high (or intrinsically infinite) dimensionality. A typical example are time series early studied by Parzen [11]. In practice, a functional datum is given 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 standard way to reduce functional data dimension is to project the functional data onto some space of functions. This approach has been extensively studied, and many papers in FDA deal with the election of the best basis [12] of the space: Fourier analysis, Wavelets, B-splines basis and Functional Principal Component Analysis (FPCA) constitute some common examples.

The key idea in our proposal is to consider each function as a point in a given function space and then to project these points onto a set of some finite-dimensional function subspaces. Then, we define appropriate kernels for those projections and combine them to obtain a kernel function valid for classification purposes. To this aim, we consider several Mercer kernels and project the original time series onto the Reproducing Kernel Hilbert Spaces (RKHS) [1,15,9,3] associated to these kernels, obtaining different finite dimensional representations of the original series.

To achieve the goal of information fusion in this context, we need to obtain a single representation for the curves from the set of different representations. To this aim we consider the natural (Euclidean) kernel matrices that arise from the

E. Bayro-Corrochano and J.-O. Eklundh (Eds.): CIARP 2009, LNCS 5856, pp. 457–464, 2009.

obtained representations and fuse them using some kernel combination technique [5]. Then, we use the Kernel Fusion [10] to obtain the function kernel to be used to classify the time series using Support Vector Machines.

This work is organized as follows. In Section 2 we show how to project a set of curves onto a RKHS generated by the eigenfunctions of a given kernel with the aid of Regularization Theory. In Section 3 we fuse the information provided by the previous projection in the frame of kernel combinations theory. We illustrate the performance of the proposed combination theory for functional data in some simulated and real experiments in Section 5 and we outline some future research lines of research in Section 6.

2 Representing Functional Data in a Reproducing Kernel Hilbert Space

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 [12].

There are several ways to introduce RKHS (see [9,1,4,15]). 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, called 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 the kernel K defined by $T_K(f) = \int_X K(\cdot, s)f(s)ds$. If we impose that $\int\int K^2(x, y)dxdy < \infty$, then T_K has a countable sequence of eigenvalues $\{\lambda_j\}$ and (orthogonal) 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 two function f and g in a function general space (that contains H_K as a subspace), they will be projected onto H_K using the operator T_K. Thus, the projections f^* and g^* will belong to the range of T_K, being $f^* = \Pi_{H_K}(f) = T_K(f)$ and $g^* = \Pi_{H_K}(g) = T_K(g)$. Applying the Spectral Theorem to T_K we get:

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

Definition 1. *Let K a kernel function with eigenfunction $\{\phi_j\}$ and T_K the linear integral operator associated to K. Consider f and g two curves in a general space Ω containing H_K. Then, we define the* ***Spectral Inner Product*** *of f and g in Ω by:*

$$\langle f, g \rangle_\Omega = \langle \Pi_{H_K}(f), \Pi_{H_K}(g) \rangle_{H_K}, \tag{2}$$

Notice that $\langle f, g\rangle_\Omega = \langle f^*, g^*\rangle_{H_K} = \sum_j \mu_j\gamma_j$ *is the standard inner product of the two elements* $f^* = \sum_j \mu_j\phi_j$ *and* $g^* = \sum_j \gamma_j\phi_j$ *in* H_K.

Next we want to obtain c_l^* for each c_l (the function corresponding to the sample functional data point $\hat{c}_l \equiv \{(\mathbf{x}_i, y_{il}) \in X \times Y\}_{i=1}^n$) in order to have a practical way to estimate the projections of the curves and to calculate (1) and (2). To find the coefficients of c_l^* (in terms of the ϕ_j in eq. (1), we use Regularization Theory to express the approximation of $\hat{c}_l$ in terms of a kernel expansion. To this aim, we seek the function c_l^* that solves the following optimization problem [4], [9]:

$$\arg\min_{c\in H_K} \frac{1}{n}\sum_{i=1}^{n} L(y_i, c(\mathbf{x}_i)) + \gamma\|c\|_K^2 . \tag{3}$$

where $\gamma > 0$, $\|c\|_K$ is the norm of the function c in H_K, $y_i = \hat{c}_i$ and $L(y_i, c(\mathbf{x}_i)) = (c(\mathbf{x}_i) - y_i)^2$. Expression (3) 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 [14], the solution c_l^* to the problem (3) exists, is unique and admits a representation of the form

$$c_l^*(\mathbf{x}) = \sum_{i=1}^{n} \alpha_{li} K(\mathbf{x}_i, \mathbf{x}), \quad \forall \mathbf{x} \in X \text{ where } \alpha_i \in \mathbb{R} . \tag{4}$$

where $\alpha_l = (\alpha_{il}, \ldots, \alpha_{nl})$ is the solution to the linear system $(\gamma n I_n + K_S)\alpha_l = y_l$ where $K_S = (K(\mathbf{x}_i, \mathbf{x}_j))_{i,j}$.

2.1 Functional Data Projections onto the Eigenfunctions Space

The particular projection we use in this work is given as follows:

Proposition 1. *Let* c *be a curve, whose sample version is* $\hat{c} = \{(\boldsymbol{x}_i, y_i) \in X \times Y\}_{i=1}^n$ *and* K *a kernel with eigenfunctions* $\{\phi_1 \ldots, \phi_d\}$ *(basis of* H_K*). Then, the projected curve* $c^*(\boldsymbol{x})$*, given by the minimization of (3), can be expressed as*

$$c_l^*(\boldsymbol{x}) = \sum_{j=1}^{d} \lambda_{lj}^* \phi_j(\boldsymbol{x}). \tag{5}$$

where λ_{lj}^* *are the weights of the projection of* $c^*(\boldsymbol{x})$ *onto the function space generated by the eigenfunctions of* K *(Span*$\{\phi_1 \ldots, \phi_d\}$*). In practice (where a finite sample is available)* λ_{lj}^* *can be estimated by*

$$\hat{\lambda}_{lj}^* = \hat{\lambda}_j \sum_{i=1}^{n} \alpha_{li}\hat{\phi}_{ji}, \tag{6}$$

being $\hat{\lambda}_j$ *the* j*th eigenvalue corresponding to the eigenvector* $\hat{\phi}_j$ *of the matrix* $K_S = (K(\boldsymbol{x}_i, \boldsymbol{x}_j))_{i,j}$, $d = \min(n, r(K_S))$*, and* α_{li} *the solution to problem (3). See [7] for further details.*

Thus we represent each curve c_l by the vector $\lambda_l = (\lambda_{l1}^*, ..., \lambda_{ld}^*)$. This representation has the nice property that is continuous in the input data [7].

3 Combining Projections via Kernel Combinations

In this section we show how to combine different representations of the curves given by the projections of the time series onto different spaces. To this aim we use some kernel combination technique to fuse the information of kernel matrices that arise from the obtained representations.

3.1 Kernels and Induced Projections

Let K a kernel function and c_1 and c_2 two time series with sample versions $\hat{c}_1$ and $\hat{c}_2$. Consider the particular curve projection onto H_K given by (3). In this case, the Spectral Inner Product of c_1 and c_2 is given by $\langle c_1, c_2 \rangle = \sum_j \lambda_{1j}^* \lambda_{2j}^*$ where λ_1^* and λ_2^* are the finite dimensional representation c_1 and c_2 in (5). Given that λ_j^* is never available we use its estimation given by eq. (6): $\sum_j \hat{\lambda}_{1j}^* \hat{\lambda}_{2j}^*$.

Definition 2. *Let $\{\hat{c}_1, \ldots, \hat{c}_m\}$ a set of sample curves and K a kernel function. the **Spectral Kernel** (SK) induced by a kernel K for two sample curves $\hat{c}_l$ and $\hat{c}_t$ is defined by*

$$\tilde{K}(\hat{c}_l, \hat{c}_t) = (\hat{\lambda}_l^*)^T \hat{\lambda}_t^*, \tag{7}$$

for $\hat{\lambda}_l^* = (\lambda_{l1}^*, \ldots, \lambda_{ld}^*)$ and $\hat{\lambda}_t^* = (\lambda_{t1}^*, \ldots, \lambda_{td}^*)$ the representation of the curves l and t estimated by eq. (6).

3.2 Combining the Representations

Let $K_1, K_2, ..., K_p$ be a set of p kernels functions inducing p different RKHS $H_{K_1}, ..., H_{K_p}$ and let $S = \{\hat{c}_1, \ldots, \hat{c}_m\}$ a labeled set of sample curves where each $\hat{c}_t$ is identified with a data set $\hat{c}_t = \{(\mathbf{x}_i, \mathbf{y}_{il}, z_l)\}_{i=1}^n$ with $z_t \in \{-1, 1\}$ (the labels of the curves). Let $\tilde{K}_{S1}, ..., \tilde{K}_{Sp}$ the p Spectral Kernels matrices (see eq. (7)) associated to the projections of the sample curves onto the spaces $H_{K_1}, ..., H_{K_p}$.

We want to combine the Spectral kernel matrices $\tilde{K}_{S1}, ..., \tilde{K}_{Sp}$ to obtain a single kernel function K^* that induces a single representation of the curves appropriate in classification problems. To this aim we select some functional kernel combination methods proposed in [9,5]. In particular we will use the Average Kernel Method (AKM), the Modified Average Kernel Method (MAKM), the Absolute Value Method (AV) and the Max-Min method. However the resulting combination matrix K^* does not need to be positive definite and does not allow directly to evaluate K_S^* at new points (where labels are not available). To fix simultaneously both problems we use the Fusion Kernel proposed in [10].

Definition 3 (Fusion Kernel). *Let $\tilde{K}_1, \ldots, \tilde{K}_p$ be a set of p kernel functions (Spectral Kernels in our case). A kernel function K is a Fusion Kernel for the set $\tilde{K}_1, \ldots, \tilde{K}_p$ when it can be expressed as*

$$K(\boldsymbol{x}, \boldsymbol{y}) = \sum_{h=1}^{d} \lambda_h \phi_h(\boldsymbol{x}) \phi_h(\boldsymbol{y}), \tag{8}$$

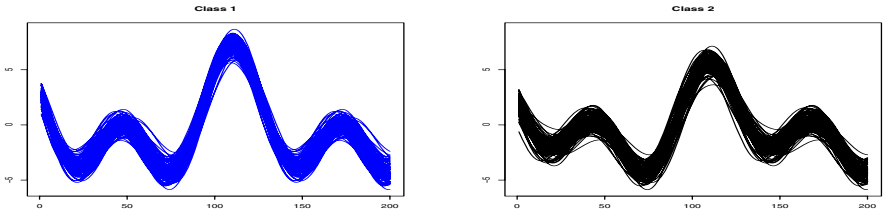


Fig. 1. Two classes of the simulated curves of the experiment

where $\{\lambda_h\} \in \mathbb{R}^+$ *and* $\phi_h \in Span\langle \underbrace{\phi_{11}, \ldots, \phi_{1d_1}}_{\tilde{K}_1}, ..., \underbrace{\phi_{p1}, \ldots, \phi_{pd_p}}_{\tilde{K}_p} \rangle$ *and* ϕ_{jr} *represents the jth eigenfunction of the rth kernel.*

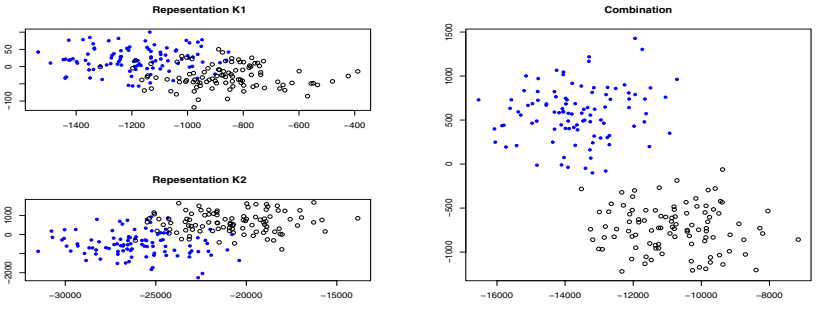
In our case, we obtain a Fusion Kernel for the matrix K^* assuming (following [10]) that every ϕ_h is defined by linear combinations of the eigenfunctions of $\tilde{K}_1, \ldots, \tilde{K}_p$. In practice, we do not know neither the eigenfunctions ϕ_h^* of the combined kernel matrix K_S^*, nor the eigenfunctions ϕ_{jr} of the Spectral kernels $\tilde{K}_{S1}, \ldots, \tilde{K}_{Sp}$. We only can compute $\hat{\gamma}_h$, the $h-th$ eigenvector of K_S^* and $\hat{\phi}_{jr}$ the jth eigenvector of the matrix K_{Sr}. However, the eigenfunctions of the kernel K^* can be approximated, up to a normalization factor, by the eigenvectors of the matrix K_S^* and the coefficients of the linear combinations of the eigenfunctions ϕ_{jr} approximating each γ_h can be approximated by a least squares projection of each $\hat{\gamma}_h$ onto the set of $\{\hat{\phi}_{jr}\}$ [10]. Finally, the eigenvalues λ_h of K^* in eq. (8) can be estimated using $\hat{\lambda}_h$, the eigenvalues of the matrix K_S^* (see [2,13] for details). However, if K_S^* is not positive definite, a transformation of them can be considered to guarantee the positive definiteness of the kernels. In this way we have all the ingredients for learning a kernel function corresponding to any kernel matrix obtained by combining the set of Spectral Kernels $\tilde{K}_{S1}, \ldots, \tilde{K}_{Sp}$.

4 Experiments

4.1 Kernels and Curves Projections: Illustrative Example

In this experiment we illustrate the behavior of our methodology in a simulated example. Consider two families of 4 dimensional curves sampled at 200 points: a) Class 1: $c(x) = \sum_{j=1}^{4} a_j \phi_j(x)$, where $a \sim N_4(\mu_1, \Sigma)$. b) Class 2: $c(x) = \sum_{j=1}^{4} b_j \phi_j(x)$, where $b \sim N_4(\mu_2, \Sigma)$ with $x \in [-5, 5]$, $\mu_1 = (2, 3, 3, 2)$, $\mu_2 = (2, 2, 2, 2)$, $\Sigma = diag(0.25, 0.25, 0.25, 0.25)$ and $\phi_1(x) = sin(x)$, $\phi_2(x) = cos(x)$, $\phi_3(x) = sin(2x)$, $\phi_4(x) = cos(2x)$. We generated 100 curves of each class. See Figure 1.

We consider two kernels to project the data onto two different RKHS via eq. (5): $K_1(\mathbf{x}, \mathbf{y}) = 0.5(\mathbf{x}^T\mathbf{y}) + 1$ and K_2 the data covariance matrix. We project the curves for both kernels by solving problem 3 for each kernel using $\gamma = 0,0001$. We



(a) Projections of the curves onto the two first eigenfunctions of kernels K_1 (top) and K_2 (down).

(b) Projections of the curves onto the two first eigenfunctions of the kernel combination

Fig. 2. Curves projections by K_1, K_2 and the AKM method

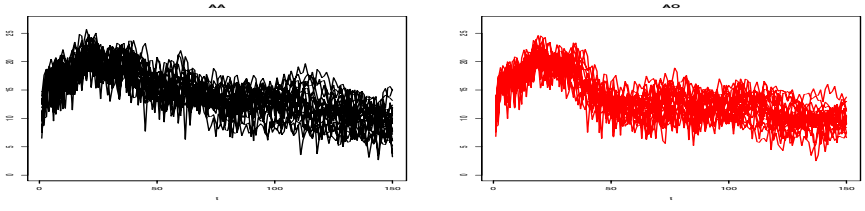


Fig. 3. Two classes of curves of the Phoneme data set

show the two first components of the projected curves in Figure 2 (a). It is apparent that none of the projections are able to separate the two classes of curves. Next we combine the Spectral kernels $\tilde{K}_1$ and $\tilde{K}_2$ resulting from both representations by using the Kernel Fusion with the MAKM procedure as combination method (see [5] for details). The projection onto the two first eigenfunctions are shown in Figure 2 (b). Now the two classes become (linearly) separable in the fusion space.

4.2 Phoneme Data Classification

The original data [6] set correspond to 2000 discretized log-periodograms of the phonemes "sh", "iy", "dcl", "aa" and "ao". Each phoneme is associated with a class of the experiment. We consider in this example those curves corresponding to the phonemes "aa" and "ao" since they present similar periodograms and are difficult to classify. A plot of 25 series of each class is shown in Figure 3.

We consider several RKHSs induced by Gaussian kernels $K_i(\mathbf{x}, \mathbf{y}) = \exp\{-\sigma_i \|\mathbf{x}-\mathbf{y}\|^2\}$ with a broad range of parameters $\sigma_i \in \{10, 7.5, 5, 2.5, 1, 0.1, 0.001\}$. We consider $\gamma = 0.001$ in eq. (3) and we project the curves using eq. (6). We estimate the Spectral kernels $\tilde{K}_i$ for $i = 1, \ldots, 7$ of the representations by using. (7) and we obtain the the Fusion Kernel in this case for the following combinations methods:

Table 1. Comparative results for the Phoneme Data after 100 runs

Method	Train Error.	Std. Dev.	Test Error	Std. Dev
Raw data	0.0682	(0.0039)	0.2606	(0.0097)
$RBF_{\sigma=10}$	0.1796	(0.0022)	0.2075	(0.0083)
$RBF_{\sigma=7.5}$	0.1787	(0.0029)	0.2037	(0.0069)
$RBF_{\sigma=5.0}$	0.1950	(0.0020)	0.2137	(0.0082)
$RBF_{\sigma=2.5}$	0.1975	(0.0019)	0.2162	(0.0068)
$RBF_{\sigma=1.0}$	0.2198	(0.0024)	0.2200	(0.0094)
$RBF_{\sigma=0.1}$	0.2242	(0.0030)	0.2243	(0.0105)
$RBF_{\sigma=0.001}$	0.2896	(0.0037)	0.2825	(0.0085)
$Fusion\ Kernel_{AKM}$	0.1868	(0.0033)	0.1906	(0.0080)
$Fusion\ Kernel_{MAKM_{\tau=1}}$	0.0000	(0.0000)	0.1881	(0.0092)
$Fusion\ Kernel_{MAXMIN}$	0.0000	(0.0000)	0.1862	(0.0069)
$Fusion\ Kernel_{AV_{\tau=1}}$	0.0000	(0.0000)	0.1875	(0.0079)
PSR	0.1866	(0.0085)	0.2033	(0.0028)
$NPCD_{derivative}$	0.2205	(0.0009)	0.3468	(0.0034)
$MPLSR_5$	0.1106	(0.0009)	0.1928	(0.0031)

AKM, MAKM, MAXMIN and AV. We use 80% of the data for training and 20% for testing and we then apply a SVM for classification (with penalization term $C = 100$) for the seven original Spectral kernels and the four Fusion Kernel combinations. We also use (for comparison purposes) two specific techniques designed to deal with functional data that have been shown to obtain very competitive results: P-spline signal regression (PSR) [8] and NPCD/MPLSR [6] with second derivative and PLS semimetrics (for dimensions 4,5,6,7 and 8). In addition, we include the results for a SVM (with linear kernel) on the raw data to compare classification results with a competitive technique that does not preprocess the data. Results are shown in Table 1.

Classifications errors in this example are large for any technique due to the overlapping of the curves. Regarding the initial RBF projections, all of them, excepting that corresponding to $\sigma = 0.001$, are able to improve the performance of a linear SVM in a particular favorable case for the linear SVM [9]. In particular, the best result for the five initial projection is given by $\sigma = 7.5$ with a 20.37% of misclassification data. However, this result is improved by the proposed fusion procedure (for all the combinations techniques) and also outperform the PSR and the MPLSR methods. Specially accurate are the results for the combinations that use labels in the fusion process as $MAKM$, $MAXMIN$ and AV that achieve errors of 18.81% and 18.62% and 18.75 respectively.

5 Conclusions

In this work we proposed a methodology for combining classifiers for functional data (time series in this paper). By considering different kernel functions we

induce different SVM classifiers and then we combine them obtaining a true kernel function with the help of a Spectral Kernel called Fusion Kernel. The idea is represent functional data by means if the eigenfunctions of kernels resulted to be interesting from the theoretical and practical point of view. The experimental results show how this methodology significantly improves the existent procedures specifically designed for time series classification. It is quite remarkable that this point of view provides different sets of basis functions in a very natural way.

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. Boser, B.E., Guyon, I., Vapnik, V.: A training algorithm for optimal margin classifiers. In: Proc. Fifth ACM Workshop on Computational Learning Theory (COLT), pp. 144–152. ACM Press, New York (1992)
4. Cucker, F., Smale, S.: On the Mathematical Foundations of Learning. Bulletin of the American Mathematical Society 39(1), 1–49 (2002)
5. de Diego, I.M.n., Moguerza, J.M., Muñoz, A.: Combining Kernel Information for Support Vector Classification. In: Roli, F., Kittler, J., Windeatt, T. (eds.) MCS 2004. LNCS, vol. 3077, pp. 102–111. Springer, Heidelberg (2004)
6. Ferraty, F., Vieu, P.: Curves discrimination: a nonparametric functional approach. Computational Statistics & Data Analysis 44, 161–173 (2003)
7. González, J., Muñoz, A.: Representing Functional Data using Support Vector Machines. In: Ruiz-Shulcloper, J., Kropatsch, W.G. (eds.) CIARP 2008. LNCS, vol. 5197, pp. 332–339. Springer, Heidelberg (2008)
8. Marx, B., Eilers, P.: Generalized linear regression on sampled signals and curves: a P-spline approach. Technometrics 41(1), 1–13 (1999)
9. Moguerza, J.M., Muñoz, A.: Support Vector Machines with Applications. Statistical Science 21(3), 322–357 (2006)
10. Muñoz, A., González, J.: Functional Learning of Kernels for Information Fusion Purposes. In: Ruiz-Shulcloper, J., Kropatsch, W.G. (eds.) CIARP 2008. LNCS, vol. 5197, pp. 277–283. Springer, Heidelberg (2008)
11. Parzen, E.: On Recent Advances in Time Series Modelling. IEEE Transactions on Automatic Control AC-19, 723–730 (1977)
12. Ramsay, J.O., Silverman, B.W.: Functional Data Analysis, 2nd edn. Springer, New York (2006)
13. Schlesinger, S.: Approximating Eigenvalues and Eigenfunctions of Symmetric Kernels. Journal of the Society for Industrial and Applied Mathematics 6(1), 1–14 (1957)
14. 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)
15. Wahba, G.: Spline Models for Observational Data. Series in Applied Mathematics, vol. 59. SIAM, Philadelphia (1990)