Classification in Hilbert Spaces with Support Vector Machines

Fabrice Rossi[1] and Nathalie Villa[2]

[1] Projet AxIS, INRIA,
Domaine de Voluceau, Rocquencourt, BP 105,
78153 Le Chesnay cedex - FRANCE
(e-mail: Fabrice.Rossi@inria.fr)
[2] Equipe GRIMM - Université Toulouse Le Mirail,
5 allées A. Machado,
31058 Toulouse cedex 1 - FRANCE
(e-mail: villa@univ-tlse2.fr)

Abstract. In many applications, input data are in fact sampled functions rather than standard high dimensional vectors. Most of the traditional data analysis tools for regression, classification and clustering have been adapted to handle functional inputs under the general name of Functional Data Analysis (FDA). In general, the major problem is to overcome the issue of infinite dimensional input. This is done by introducing regularity constraints on the studied functions, thanks to penalization or to projection on finite dimensional functional spaces.

Support Vector Machine (SVM) are large margin classifier tools that have the interesting property of being less sensitive to the curse of dimensionality than other tools. On the contrary, they are based on implicit non linear mappings of the considered data into high dimensional spaces (sometimes with infinite dimension) thanks to kernel functions.

In this paper, we investigate the use of Support Vector Machine for functional data analysis. We define simple kernels that take into account the functional nature of the data and lead to consistent classification. Experiments conducted on real world data emphasize the benefit of taking into account some functional aspects of the problems.
Keywords: Functional Data Analysis, Support Vector Machine, Classification.

1 Introduction

This paper deals with functional classification: let (X, Y) be a pair of random variables in which Y takes values in $\{-1; 1\}$ and X in a functional space. Y is the label (the class) associated to X. The goal of classification is to predict the value of Y given an observed value for X. The difficulty in functional data analysis [Ramsay and Silverman, 1997], compared to the traditionnal setting, is that X does not take values in $\mathbb{R}^d$ but in a functional space.

In this paper, we investigate how Support Vector Machine (SVM) can be used for functional data classification. The paper is organized as follows: Section 2 explains why functional SVM leads to particular problems and

proposes solutions to overcome them. Section 3 develops several functional kernels and explains how some of them lead to consistent classifier. Finally, Section 5 illustrates the various approaches on real data sets.

2 Support Vector Machine For FDA

2.1 Hard margin functional SVM

We assume given a learning set, i.e. N examples $(x_1, y_1), \ldots, (x_N, y_N)$ which are i.i.d. realizations of (X, Y). As explained before, X is a function valued random variable. More formally, X takes its values in a separable Hilbert space $\mathcal{X}$, for example a subspace of $L^2(\mu)$ where μ denotes a finite Borel measure on $\mathbb{R}$. We denote $\langle ., . \rangle$ the inner product of $\mathcal{X}$.

The principle of SVM is to perform an affine discrimination of the observations with the largest margin as possible, that is to find a function $w \in \mathcal{X}$ with a minimum norm and a real value b, such that $y_i(\langle w, x_i \rangle + b) \geq 1$ for all i. The classification rule associated to (w, b) is simply $\phi(x) = \text{sign}(\langle w, x \rangle + b)$. We therefore request the rule to have zero error on the learning set.

In functional spaces, it is always possible to find such a discrimination, provided the $(x_i)_{1 \leq i \leq N}$ are in general position, i.e. provided they span a vector space of dimension N. However it is well known that the obtained classification rule do not behave in a satisfactory way unless a regularization method is used (see [Hastie and Mallows, 1993], [Marx and Eilers, 1996], [Ramsay and Silverman, 1997] and [Cardot *et al.*, 1999]).

2.2 Soft margin functional SVM

While SVM introduces a form of regularization by looking for large margin (i.e., minimal norm for w), additional regularization can be obtained by solving the following optimization problem:

$$\begin{array}{ll} (P_C) & \min_{w,b,\xi} \langle w, w \rangle + C \sum_{i=1}^N \xi_i, \\ & \text{subject to } y_i(\langle w, x_i \rangle + b) \geq 1 - \xi_i, \\ & \qquad \xi_i \geq 0, \text{ for all } i = 1, \ldots, N, \end{array}$$

for an appropriate $C \geq 0$. Using the slack variables ξ_i allows to relax the very strong condition that the classification rule should make no error on the learning set. It is well known (see e.g., [Hastie *et al.*, 2004]) that this form of regularization is needed to achieve good performances for classification in high dimensional spaces.

In order to solve this problem, we use results from [Chih-Jen, 2001] that apply to any Hilbert space. Problem (P_C) is indeed equivalent to the dual optimization problem:

$$\begin{array}{ll} (D_C) & \min_{\alpha} \sum_{i=1}^N \alpha_i - \sum_{i,j=1}^N \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle, \\ & \text{subject to } \sum_{i=1}^N \alpha_i y_i = 0 \text{ and } 0 \leq \alpha_i \leq C \text{ for all } i = 1, \ldots, N. \end{array}$$

The advantage of (D_C) versus (P_C) in the infinite dimensional context is that the optimization problem (D_C) has to be solved in $\mathbb{R}^N$ whereas (P_C) needs an optimization procedure in $\mathcal{X}$. Moreover inner products in functional spaces such as $L^2(\mu)$ are easy to approximate using classical quadrature or Monte Carlo methods. Finaly, the classification rule is obtained as $\phi(x) = \text{sign}(\sum_{i=1}^{N} y_i \alpha_i \langle x_i, x \rangle + b)$ which is only based on inner products. In practice, this means that any SVM software can be used to provide functional classification as long as inner products can be calculated and used in the software.

It should be noted that C is a free parameter. It has therefore to be chosen so has to provide good performances. We will provide a possible solution in section 4.1.

3 Functional kernels

3.1 Kernels for SVM

A major difference between standard multivariate data and functional data is that the former are seldom linearly separable whereas the latter often are. In finite dimensional settings, this motivates the use of kernels to replace the inner product that is used in problem (D_c). A kernel corresponds to an implicit mapping from the input space to another feature space. In general this feature space has a high dimension so that the data become linearly separable in it. Thanks to the dual formulation of the SVM optimization problem, the implicit mapping is not calculated: everything is based on the kernel.

For functional data, the use of kernels might seem worthless. However, despite the regularization provided by using slack variables, it happens in practice for linear functional SVM to have very bad performances. A possible solution consists in using functional transformation and functional kernels, as proposed in this section.

3.2 Using an orthogonal basis

A natural functional kernel can be constructed thanks to the general functional classification framework proposed in [Biau *et al.*, 2005]. The methods proceeds as follows:

1. choose a complete orthonormal system of $\mathcal{X}$, $\{\Psi_j\}_{j\geq 1}$, and express each observation x_i as a series expansion $x_i = \sum_{j\geq 1} x_{ij} \Psi_j$;
2. approximate each observation x_i by the sum $\sum_{j=1}^{d} x_{ij} \Psi_j$;
3. perform a classical $\mathbb{R}^d$ SVM on the coefficients $\mathbf{x}_i^{(d)} = (x_{i1}, \ldots, x_{id}) \in \mathbb{R}^d$ for all $i = 1, \ldots, N$.

This procedure is equivalent to working with a functional kernel which can be written as

$$\mathcal{K}_d(x, x') = K(\mathcal{P}_d(x), \mathcal{P}_d(x'))$$

where $\mathcal{P}_d$ denotes the projection onto the the space spanned by $\{\Psi_j\}_{j=1,\dots,d}$ and K is any standard SVM kernel. Of course, d has to be chosen appropriately. As recalled in section 4.1, [Biau *et al.*, 2005] proposes to use a split sample approach.

3.3 Using a B-Spline basis

Another way of choosing a projection space consists in using spline spaces and their B-spline bases. Results from [Biau *et al.*, 2005] are still applicable, but with major restriction. Indeed, a B-spline basis is not a basis of L^2: it only spans a subspace of L^2. Nevertheless, they perform efficiently in practice.

An interesting property of B-spline bases if they can be use to provide additional transformation on the input data: using a B-Spline expansion, an estimation of $x^{(q)}$, the q-th derivative of x, can be easily obtained. Then any kernel can be used on the derivatives. This method allows to focus on some particular aspects of the underlying functions, such as the curvature for the second derivative. It is well known that in some application domain such as spectrometry, such kind of features might be more interesting than the original curves. We give in Section 5.3 an application of this approach.

4 Consistency of functional SVM

4.1 Choice of the parameters

Performing a functional SVM leads to choose three types of parameters:

1. parameters due to the functional pre-processing: d, the dimension of the projection if we use a orthogonal basis as in section 3.2 or the order of the B-Splines basis, the number of knots and the order q of the derivative(s) chosen in the case of the pre-processing described in section 3.3;
2. C, the regularization parameter of the SVM (see section 2.2);
3. K, which is indeed the kernel: we can both choose the type of kernel (linear, gaussian, ...) but also the parameter of this kernel such as σ for the gaussian kernel $K(x, x') = e^{-\|x-x'\|^2/\sigma}$.

In order to select these parameters, we follow [Biau *et al.*, 2005] and use a data splitting device. To do that, let us introduce some notations: a denotes the parameters that we have to chosen in a set $\mathcal{A}$ of relevant parameters and $\mathcal{P}$ the preprocessing performed on the original data set. The data are then split into two sets. First, for a fixed value of the parameters, a, a training set $\{(x_i, y_i), i = 1, \dots, l\}$ is used to calculate the SVM classification rule

$\phi_a^l = \text{sign}(\sum_{i=1}^{l} \alpha_i^* y_i K(\mathcal{P}(.), \mathcal{P}(x_i)) + b^*)$ where $(\{\alpha_i^*\}_i, b^*)$ are the solution of (D_C) in which we replace the classical dot product by $K \circ \mathcal{P}$. Then a validation set $\{(x_i, y_i), i = l+1, \ldots, N\}$ is used to select a optimally in $\mathcal{A}$:

$$a^* = \arg\min_{a \in \mathcal{A}} \left\{ \hat{L}(\phi_a^l) + \frac{\lambda_a}{\sqrt{N-l}} \right\}.$$

where $\hat{L}(\phi_a^l) = \frac{1}{m} \sum_{i=l+1}^{N} \mathbf{1}_{\{\phi_a^l(x_i) \neq y_i\}}$ and $\frac{\lambda_a}{\sqrt{N-l}}$ is a penalty term.

4.2 Consistency

We now restrict ourselves to the case of the functional kernels of section 3.2. Then, as pointed out by [Biau *et al.*, 2005], a necessary and sufficient condition of consistency for the procedure described in sections 3.2 and 4.1 is that classical SVM are consistent in $\mathbb{R}^d$. [Steinwart, 2002] shows the universal consistency of some SVMs when two conditions are fulfilled: the input data must belong to a compact subset of $\mathbb{R}^d$ and the regularization parameter for N observations must be equal to $C_N = N^{\beta-1}$ (see Corollary 1 of [Steinwart, 2002]). This consistency result holds as long as the kernel used to perform it is *universal*; that is : if Φ is the feature map of the kernel, then the set of all the functions of the form $\langle w, \Phi(.) \rangle$ has to be dense in the set of all continuous functions defined on the considered compact subset. In particular, the gaussian kernel with any $\sigma > 0$ is universal for all compact subsets of $\mathbb{R}^d$.

Therefore, for this procedure, the choice of $a = (d, C, K)$ leads to a consistent classifier provinding some simple facts: for any fixed dimension d, K has to be chosen in a finite set $\mathcal{K}_d$ which contains, at least, one universal kernel. C can be chosen in a finite grid search (as this is the case in our applications) but recent progresses (see [Hastie *et al.*, 2004]) allows to choose C in an interval of the form $\mathcal{I}_d = [0; \mathcal{C}_d]$ by an automatic recurrent procedure.

The consistency result of [Biau *et al.*, 2005] is obtained for a k-nn classifier but, as stated in the paper, the result can be extended to any classifier. When choosing C in a infinite set, an adaptation of the proof is needed. As the original proof is constructed thanks to an oracle inequality that gives an upper bound for $EL(\phi_{d^*,C^*,K^*}) - L^*$ in finite dimension (L^* denotes the Bayes error), we have to obtain a similar oracle inequality: this can be done by the use of the shatter coefficient of a particular class of linear classifiers which provides the behavior of the classification rule on a set of $N - l$ points (see [Devroye *et al.*, 1996]). A limitation of SVM that does not appear in [Biau *et al.*, 2005] for k-nn, is that the input functions must belong to a compact subset of the functional space.

5 Applications

5.1 Speech recognition in very high dimensional space

We compare SVM to k-nn by applying exactly the procedure described in [Biau *et al.*, 2005] to the data used in the paper. The only difference is the

replacement of the k-nn classifier by a regular SVM. The problem considered in [Biau *et al.*, 2005] consists in classifying speech samples. There are three two classes problems: classifying "yes" against "no", "boat" against "goat" and "sh" against "ao". For each problem, we have 100 functions. Each function is described by a vector in $\mathbb{R}^{8192}$. Performances of the algorithms are obtained thanks to a leave-one-out procedure: 99 functions are used as the learning set (to which the split sample procedure is applied to choose the parameters) and the remaining function provide a test example. We use the Fourier functional basis. We report the percentage of error for each problem in the following table:

Problem	k-nn	QDA	Gaussian SVM	linear SVM
yes/no	10%	7%	10%	58%
boat/goat	21%	35%	8%	46%
sh/ao	16%	32%	12%	47%

The first two columns have been reproduced from [Biau *et al.*, 2005] (QDA corresponds to Quadratic Discriminant Analysis). The "Gaussian SVM" column corresponds to the functional kernel obtained thanks to the projection of the Fourier basis combined to a Gaussian kernel in $\mathbb{R}^d$. The "linear SVM" corresponds to the direct application of the procedure described in 2.2, without any prior projection. In general the functional kernel give very satisfactory results, whereas the direct linear approach obtain extremely bad results (they corresponds to a random classification). This shows that the regularization provided by the slack variables is not adapted to functional data, a fact that was already known in the context of linear discriminant analysis [Hastie *et al.*, 1995].

The functional SVM performs in general better than k-nn and QDA, but the training time of the methods are not comparable. Indeed, solving problem (D_C) can cost up to $O(N^3)$ operations, whereas there is no training time for k-nn.

5.2 Using wavelet basis

In order to investigate the limitation of the direct use of the linear SVM, we have applied them to another speech recognition problem. We studied a part of TIMIT database which was investigated in [Hastie *et al.*, 1995]. The data are log-periodograms corresponding to recording phonemes of 32 ms duration. We have chosen to restrict ourselves to classifying "aa" against "ao", because this is the most difficult sub-problem in the database. The database is a multi-speaker database. Each speaker (325 in the training set and 112 in the test set) is recorded at a 16-kHz sampling rate; and we retain only the first 256 frequencies. We have 519 examples for "aa" in the training set (759 for "ao") and 176 in the test set (263 for "ao"). We use the split sample approach to choose the parameters on the training set (50% of the

training examples are used for validation) and we report the classification error on the test set. The projection basis is here a hierarchical wavelet basis (see e.g., [Mallat, 1989]). We obtain the following results:

Functional Gaussian SVM	Functional linear SVM	Linear SVM
22%	19.4%	20%

It appears that functional kernels are not as useful here as in the previous example, as linear SVM applied directly to the discretized functions (in $\mathbb{R}^{256}$) performs as well as linear SVM on the wavelet coefficients. A natural explanation is that the actual dimension of the input space (256) is smaller than the number of learning examples (1278) which means that evaluating the optimal coefficients of the SVM is less difficult than in the previous example. Therefore, the additional regularization provided by the projection is not really useful in this context.

5.3 Spectrometric data set

The data presented in this section are 215 near infrared spectra of a meat sample recorded on a Tecator Infrared Food and Feed Analyser[1]. The classification problem consists in separating meat samples with a high fat content (more than 20%) from sample with a low fat content (less than 20%). It is well known that in some spectrometric problem, the curvature of the spectrum is more relevant for the prediction of the sample content than the spectrum itself. This drives us to construct a classifier based on the curvature of the spectra i.e. on the second derivative as explained in section 3.3.

We then decide to compare: a linear and a gaussian kernel performed on the original data and a linear and a gaussian kernel on the second derivatives. The training set contains 120 spectra (randomly chosen) and the testing set 95 spectra. The parameters (C and σ for the gaussian kernel) are chosen using a 10-fold cross validation procedure rather than a simple cross validation procedure. The following table gives the performances of the various methodologies:

Kernel	Learning set error rate	Test set error rate
Linear	0.83%	2.11%
Gaussian	0%	4.21%
Linear on second derivatives	0%	0%
Gaussian on second derivatives	0.83%	1.05%

It appear that the functional pre-processing slightly improves the results: in both linear and gaussian kernels, the use of the second derivatives introduces a kind of expert knowledge and overcomes the limitation of the original kernel. This is specially the case for the gaussian kernel which is norm dependant and is then dominated by the mean value of the spectra (which is not a good feature for spectrometric problems as we already said).

[1] available on statlib: `http://lib.stat.cmu.edu/datasets/tecator`

6 Conclusion

We have proposed in this paper functional kernels that provide consistent classification in Hilbert spaces with Support Vector Machines. When the considered functions are represented by very high dimensional vectors, projection based kernels provide regularization that enhance SVM classification rates. In other contexts, transformation based kernels allow to integrate expert knowledge in the SVM.

References

[Biau *et al.*, 2005]Gérard Biau, Florentina Bunea, and Marten Wegkamp. Functional classification in hilbert spaces. *IEEE Transactions on Information Theory*, 2005. To be published.

[Cardot *et al.*, 1999]Hervé Cardot, Frédéric Ferraty, and Pascal Sarda. Functional linear model. *Statist. & Prob. Letters*, 45:11–22, 1999.

[Chih-Jen, 2001]L. Chih-Jen. Formulation of support vector machines: a note from an optimization point of view. *Neural Computation*, 2(13):307–317, 2001.

[Devroye *et al.*, 1996]L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer Verlag, New York, 1996.

[Hastie and Mallows, 1993]T. Hastie and C. Mallows. A discussion of "a statistical view of some chemometrics regression tools" by i.e. frank and j.h. friedman. *Technometrics*, 35:140–143, 1993.

[Hastie *et al.*, 1995]T. Hastie, A. Buja, and R. Tibshirani. Penalized discriminant analysis. *Annals of Statistics*, 23:73–102, 1995.

[Hastie *et al.*, 2004]Trevor Hastie, Saharon Rosset, Robert Tibshirani, and Ji Zhu. The entire regularization path for the support vector machine. *Journal of Machine Learning Research*, 5:1391–1415, October 2004.

[Mallat, 1989]Stéphane Mallat. Multiresolution approximation and wavelet orthonormal bases of l2. *Transaction of the American Mathematical Society*, 315:69–87, September 1989.

[Marx and Eilers, 1996]B. D. Marx and P. H. Eilers. Generalized linear regression on sampled signals with penalized likelihood. In R. Hatzinger A. Forcina, G. M. Marchetti and G. Galmacci, editors, *Statistical Modelling. Proceedings of the 11th International workshop on Statistical Modelling*, Orvietto, 1996.

[Ramsay and Silverman, 1997]Jim Ramsay and Bernard Silverman. *Functional Data Analysis*. Springer Series in Statistics. Springer Verlag, June 1997.

[Steinwart, 2002]I. Steinwart. Support vector machines are universally consistent. *J. Complexity*, 18:768–791, 2002.