# Consistency of functional learning methods based on derivatives

Fabrice Rossi, Nathalie Villa-Vialaneix

# Consistency of Functional Learning Methods Based on Derivatives

Fabrice Rossi[a], Nathalie Villa-Vialaneix[b,c,*]

[a] *Télécom ParisTech, LTCI - UMR CNRS 5141, France*
[b] *IUT de Perpignan (Dpt STID, Carcassonne), Université de Perpignan Via Domitia, France*
[c] *Institut de Mathématiques de Toulouse, Université de Toulouse, France*

**Abstract**

In some real world applications, such as spectrometry, functional models achieve better predictive performances if they work on the derivatives of order $m$ of their inputs rather than on the original functions. As a consequence, the use of derivatives is a common practice in functional data analysis, despite a lack of theoretical guarantees on the asymptotically achievable performances of a derivative based model. In this paper, we show that a smoothing spline approach can be used to preprocess multivariate observations obtained by sampling functions on a discrete and finite sampling grid in a way that leads to a consistent scheme on the original infinite dimensional functional problem. This work extends Mas and Pumo (2009) to nonparametric approaches and incomplete knowledge. To be more precise, the paper tackles two difficulties in a nonparametric framework: the information loss due to the use of the derivatives instead of the original functions and the information loss due to the fact that the functions are observed through a discrete sampling and are thus also unperfectly known: the use of a smoothing spline based approach solves these two problems. Finally, the proposed approach is tested on two real world datasets and the approach is experimentaly proven to be a good solution in the case of noisy functional predictors.

*Keywords:* Functional Data Analysis, Consistency, Statistical learning, Derivatives, SVM, Smoothing splines, RKHS, Kernel

*Corresponding author.
*Email addresses:* Fabrice.Rossi@telecom-paristech.fr (Fabrice Rossi), nathalie.villa@math.univ-toulouse.fr (Nathalie Villa-Vialaneix)

## 1. Introduction

As the measurement techniques are developping, more and more data are high dimensional vectors generated by measuring a continuous process on a discrete sampling grid. Many examples of this type of data can be found in real world applications, in various fields such as spectrometry, voice recognition, time series analysis, etc.

Data of this type should not be handled in the same way as standard multivariate observations but rather analysed as *functional* data: each observation is a function coming from an input space with infinite dimension, sampled on a high resolution sampling grid. This leads to a large number of variables, generally more than the number of observations. Moreover, functional data are frequently smooth and generate highly correlated variables as a consequence. Applied to the obtained high dimensional vectors, classical statistical methods (e.g., linear regression, factor analysis) often lead to ill-posed problems, especially when a covariance matrix has to be inverted (this is the case, e.g., in linear regression, in discriminant analysis and also in sliced inverse regression). Indeed, the number of observed values for each function is generally larger than the number of functions itself and these values are often strongly correlated. As a consequence, when these data are considered as multidimensional vectors, the covariance matrix is ill-conditioned and leads to unstable and unaccurate solutions in models where its inverse is required. Thus, these methods cannot be directly used. During past years, several methods have been adapted to that particular context and grouped under the generic name of Functional Data Analysis (FDA) methods. Seminal works focused on linear methods such as factorial analysis (Deville (1974); Dauxois and Pousse (1976); Besse and Ramsay (1986); James et al. (2000), among others) and linear models Ramsay and Dalzell (1991); Cardot et al. (1999); James and Hastie (2001); a comprehensive presentation of linear FDA methods is given in Ramsay and Silverman (1997, 2002). More recently, nonlinear functional models have been extensively developed and include generalized linear models James (2002); James and Silverman (2005), kernel nonparametric regression Ferraty and Vieu (2006), Functional Inverse Regression Ferré and Yao (2003), neural networks Rossi and Conan-Guez (2005); Rossi et al. (2005), $k$-nearest neighbors Biau et al. (2005); Laloë (2008), Support Vector Machines (SVM), Rossi and Villa (2006), among a

very large variety of methods.

In previous works, numerous authors have shown that the derivatives of the functions lead sometimes to better predictive performances than the functions themselves in inference tasks, as they provide information about the shape or the regularity of the function. In particular applications such as spectrometry Ferraty and Vieu (2006); Rossi et al. (2005); Rossi and Villa (2006), micro-array data Dejean et al. (2007) and handwriting recognition Williams et al. (2006); Bahlmann and Burkhardt (2004), these characteristics lead to accurate predictive models. But, on a theoretical point of the view, limited results about the effect of the use of the derivatives instead of the original functions are available: Mas and Pumo (2009) studies this problem for a linear model built on the first derivatives of the functions. In the present paper, we also focus on the theoretical relevance of this common practice and extend Mas and Pumo (2009) to nonparametric approaches and incomplete knowledge.

More precisely, we address the problem of the estimation of the conditional expectation $\mathbb{E}(Y|X)$ of a random variable $Y$ given a functional random variable $X$. $Y$ is assumed to be either real valued (leading to a regression problem) or to take values in $\{-1, 1\}$ (leading to a binary classification problem). We target two theoretical difficulties. The first difficulty is the potential information loss induced by using a derivative instead of the original function: when one replaces $X$ by its order $m$ derivative $X^{(m)}$, consistent estimators (such as kernel models Ferraty and Vieu (2006)) guarantee an asymptotic estimation of $\mathbb{E}\left(Y|X^{(m)}\right)$ but cannot be used directly to address the original problem, namely estimating $\mathbb{E}(Y|X)$. This is a simple consequence of the fact that $X \mapsto X^{(m)}$ is not a one to one mapping. The second difficulty is induced by sampling: in practice, functions are never observed exactly but rather, as explained above, sampled on a discrete sampling grid. As a consequence, one relies on approximate derivatives, $\widehat{X}_\tau^{(m)}$ (where $\tau$ denotes the sampling grid). This approach induces even more information loss with respect to the underlying functional variable $X$: in general, a consistent estimator of $\mathbb{E}\left(Y|\widehat{X}_\tau^{(m)}\right)$ will not provide a consistent estimation of $\mathbb{E}(Y|X)$ and the optimal predictive performances for $Y$ given $\widehat{X}_\tau^{(m)}$ will be lower than the optimal predictive performances for $Y$ given $X$.

We show in this paper that the use of a smoothing spline based approach solves both problems. Smoothing splines are used to estimate the functions from their sampled version in a convergent way. In addition, properties of

splines are used to obtain estimates of the derivatives of the functions with no induced information loss. Both aspects are implemented as a preprocessing step applied to the multivariate observations generated via the sampling grid. The preprocessed observations can then be fed into any finite dimensional consistent regression estimator or classifier, leading to a consistent estimator for the original infinite dimensional problem (in real world applications, we instantiate the general scheme in the particular case of kernel machines Shawe-Taylor and Cristianini (2004)).

The remainder of the paper is organized as follows: Section 2 introduces the model, the main smoothness assumption and the notations. Section 3 recalls important properties of spline smoothing. Section 4 presents approximation results used to build a general consistent classifier or a general consistent regression estimator in Section 5. Finally, Section 6 illustrates the behavior of the proposed method for two real world spectrometric problems. The proofs are given at the end of the article.

## 2. Setup and notations

### *2.1. Consistent classifiers and regression functions*

We consider a pair of random variables $(X, Y)$ where $X$ takes values in a functional space $\mathcal{X}$ and $Y$ is either a real valued random variable (regression case) or a random variable taking values in $\{-1, 1\}$ (binary classification case). From this, we are given a learning set $S_n = \{(X_i, Y_i)\}_{i=1}^n$ of $n$ independent copies of $(X, Y)$. Moreover, the functions $X_i$ are not entirely known but sampled according to a non random sampling grid of finite length, $\tau_d = (t_l)_{l=1}^{|\tau_d|}$: we only observe $\mathbf{X}_i^{\tau_d} = (X_i(t_1), \dots X_i(t_{|\tau_d|}))^T$, a vector of $\mathbb{R}^{|\tau_d|}$ and denote $S_{n,\tau_d}$ the corresponding learning set. Our goal is to construct:

1. *in the binary classification case*: a classifier, $\phi_{n,\tau_d}$, whose misclassification probability
$$L(\phi_{n,\tau_d}) = \mathbb{P}\left(\phi_{n,\tau_d}(\mathbf{X}^{\tau_d}) \neq Y\right)$$
asymptotically reaches the Bayes risk
$$L^* = \inf_{\phi:\mathcal{X}\to\{-1,1\}} \mathbb{P}\left(\phi(X) \neq Y\right)$$
i.e., $\lim_{|\tau_d|\to+\infty} \lim_{n\to+\infty} \mathbb{E}\left(L(\phi_{n,\tau_d})\right) = L^*$ ;

2. *in the regression case*: a regression function, $\phi_{n,\tau_d}$, whose $L^2$ error

$$L(\phi_{n,\tau_d}) = \mathbb{E}\left([\phi_{n,\tau_d}(\mathbf{X}^{\tau_d}) - Y]^2\right)$$

asymptotically reaches the minimal $L^2$ error

$$L^* = \inf_{\phi:\mathcal{X}\to\mathbb{R}} \mathbb{E}\left([\phi(\mathbf{X}^{\tau_d}) - Y]^2\right)$$

i.e., $\lim_{|\tau_d|\to+\infty} \lim_{n\to+\infty} L(\phi_{n,\tau_d}) = L^*$.

This definition implicitly requires $\mathbb{E}(Y^2) < \infty$ and as a consequence, corresponds to a $L^2$ convergence of $\phi_{n,\tau_d}$ to the conditional expectation $\phi^* = \mathbb{E}(Y|X)$, i.e., to $\lim_{|\tau_d|\to+\infty} \lim_{n\to+\infty} \mathbb{E}\left([\phi_{n,\tau_d}(\mathbf{X}^{\tau_d}) - \phi^*(X)]^2\right) = 0$.

Such $\phi_{n,\tau_d}$ are said to be *(weakly) consistent* Devroye et al. (1996); Györfi et al. (2002). We have deliberately used the same notations for the (optimal) predictive performances in both the binary classification and the regression case. We will call $L^*$ the Bayes risk even in the case of regression. Most of the theoretical background of this paper is common to both the regression case and the classification case: the distinction between both cases will be made only when necessary.

As pointed out in the introduction, the main difficulty is to show that the performances of a model built on the $\mathbf{X}_i^{\tau_d}$ asymptotically reach the best performance achievable on the original functions $X_i$. In addition, we will build the model on derivatives estimated from the $\mathbf{X}_i^{\tau_d}$.

### *2.2. Smoothness assumption*

Our goal is to leverage the functional nature of the data by allowing differentiation operators to be applied to functions prior their submission to a more common classifier or regression function. Therefore we assume that the functional space $\mathcal{X}$ contains only differentiable functions. More precisely, $\mathcal{X}$ is the Sobolev space $\mathcal{H}^m = \left\{h \in L^2([0,1]) \mid \forall j = 1,\ldots,m,\ D^j h \text{ exists in the weak sense, and } D^m h \in L^2([0,1])\right\}$, where $D^j h$ is the *j-th* derivative of $h$ (also denoted by $h^{(j)}$) and for an integer $m \geq 1$. Of course, by a straightforward generalization, any bounded interval can be considered instead of $[0,1]$.

To estimate the underlying functions $X_i$ and their derivatives from sampled data, we rely on smoothing splines. More precisely, let us consider a deterministic function $x \in \mathcal{H}^m$ sampled on the aforementioned grid. A smoothing spline estimate of $x$ is the solution, $\widehat{x}_{\lambda,\tau_d}$, of

$$\arg\min_{h\in\mathcal{H}^m} \frac{1}{|\tau_d|}\sum_{l=1}^{|\tau_d|}(x(t_l)-h(t_l))^2 + \lambda\int_{[0,1]}(h^{(m)}(t))^2dt, \tag{1}$$

where $\lambda$ is a regularization parameter that balances interpolation error and smoothness (measured by the $L^2$ norm of the $m$-$th$ derivative of the estimate). The goal is to show that a classifier or a regression function built on $\widehat{X}^{(m)}_{\lambda,\tau_d}$ is consistent for the original problem (i.e., the problem defined by the pair $(X,Y)$): this means that using $\widehat{X}^{(m)}_{\lambda,\tau_d}$ instead of $X$ has no dramatic consequences on the accuracy of the classifier or of the regression function. In other words, asymptotically, no information loss occurs when one replaces $X$ by $\widehat{X}^{(m)}_{\lambda,\tau_d}$.

The proof is based on the following steps:

1. First, we show that building a classifier or a regression function on $\widehat{X}^{(m)}_{\lambda,\tau_d}$ is approximately equivalent to building a classifier or a regression function on $\mathbf{X}^{\tau_d} = (X(t_l))_{l=1}^{|\tau_d|}$ using a specific metric. This is done by leveraging the Reproducing Kernel Hilbert Space (RKHS) structure of $\mathcal{H}^m$. This part serves one main purpose: it provides a solution to work with estimation of the derivatives of the original function in a way that preserves all the information available in $\mathbf{X}^{\tau_d}$. In other words, the best predictive performances for $Y$ theoretically available by building a multivariate model on $\mathbf{X}^{\tau_d}$ are equal to the best predictive performances obtained by building a functional model on $\widehat{X}^{(m)}_{\lambda,\tau_d}$.

2. Then, we link $\mathbb{E}\left(Y|\widehat{X}_{\lambda,\tau_d}\right)$ with $\mathbb{E}\left(Y|X\right)$ by approximation results available for smoothing splines. This part of the proof handles the effects of sampling.

3. Finally, we glue both results via standard $\mathbb{R}^{|\tau_d|}$ consistency results.

## 3. Smoothing splines and differentiation operators

### *3.1. RKHS and smoothing splines*

As we want to work on derivatives of functions from $\mathcal{H}^m$, a natural inner product for two functions of $\mathcal{H}^m$ would be $(u,v) \to \int_0^1 u^{(m)}(t)v^{(m)}(t)dt$. However, we prefer to use an inner product of $\mathcal{H}^m$ ($\int_0^1 u^{(m)}(t)v^{(m)}(t)dt$ only induces a semi-norm on $\mathcal{H}^m$) because, as will be shown later, such an inner product is related to an inner product between the sampled functions considered as vectors of $\mathbb{R}^{|\tau_d|}$.

This can be done by decomposing $\mathcal{H}^m$ into $\mathcal{H}^m = \mathcal{H}_0^m \oplus \mathcal{H}_1^m$ Kimeldorf and Wahba (1971), where $\mathcal{H}_0^m = \mathrm{Ker}D^m = \mathbb{P}^{m-1}$ (the space of polynomial functions of degree less or equal to $m-1$) and $\mathcal{H}_1^m$ is an infinite dimensional subspace of $\mathcal{H}^m$ defined via $m$ boundary conditions. The boundary conditions are given by a full rank linear operator from $\mathcal{H}^m$ to $\mathbb{R}^m$, denoted $B$, such that $\mathrm{Ker}B \cap \mathbb{P}^{m-1} = \{0\}$. Classical examples of boundary conditions include the case of "natural splines" (for $m=2$, $h(0)=h(1)=0$) and constraints that target only the first values of $h$ and its derivatives at a fixed position, for instance the conditions: $h(0) = \ldots = h^{(m-1)}(0) = 0$. Other boundary conditions can be used Berlinet and Thomas-Agnan (2004); Besse and Ramsay (1986); Craven and Wahba (1978), depending on the application.

Once the boundary conditions are fixed, an inner product on both $\mathcal{H}_0^m$ and $\mathcal{H}_1^m$ can be defined:

$$\langle u, v\rangle_1 = \langle D^m u, D^m v\rangle_{L^2} = \int_0^1 u^{(m)}(t)v^{(m)}(t)dt$$

is an inner product on $\mathcal{H}_1^m$ (as $h \in \mathcal{H}_1^m$ and $D^m h \equiv 0$ give $h \equiv 0$). Moreover, if we denote $B = (B^j)_{j=1}^m$, then $\langle u, v\rangle_0 = \sum_{j=1}^m B^j u B^j v$ is an inner product on $\mathcal{H}_0^m$. We obtain this way an inner product on $\mathcal{H}^m$ given by

$$\begin{aligned}\langle u, v\rangle_{\mathcal{H}^m} &= \int_0^1 u^{(m)}(t)v^{(m)}(t)dt + \sum_{j=1}^m B^j u B^j v \\ &= \langle \mathcal{P}_1^m(u), \mathcal{P}_1^m(v)\rangle_1 + \langle \mathcal{P}_0^m(u), \mathcal{P}_0^m(v)\rangle_0\end{aligned}$$

where $\mathcal{P}_i^m$ is the projector on $\mathcal{H}_i^m$.

Equipped with $\langle ., .\rangle_{\mathcal{H}^m}$, $\mathcal{H}^m$ is a Reproducing Kernel Hilbert Space (RKHS, see e.g. Berlinet and Thomas-Agnan (2004); Heckman and Ramsay (2000); Wahba (1990)). More precisely, it exists a kernel $k : [0,1]^2 \to \mathbb{R}$ such

that, for all $u \in \mathcal{H}^m$ and all $t \in [0,1]$, $\langle u, k(t,.)\rangle_{\mathcal{H}^m} = u(t)$. The same occurs for $\mathcal{H}_0^m$ and $\mathcal{H}_1^m$ which respectively have reproducing kernels denoted by $k_0$ and $k_1$. We have $k = k_0 + k_1$.

In the most common cases, $k_0$ and $k_1$ have already been explicitly calculated (see e.g., Berlinet and Thomas-Agnan (2004), especially chapter 6, sections 1.1 and 1.6.2). For example, for $m \geq 1$ and the boundary conditions $h(0) = h'(0) = \ldots = h^{(m-1)}(0) = 0$, we have:

$$k_0(s,t) = \sum_{k=0}^{m-1} \frac{t^k s^k}{(k!)^2}.$$

and

$$k_1(s,t) = \int_0^1 \frac{(t-w)_+^{m-1}(s-w)_+^{m-1}}{(m-1)!^2}\, dw.$$

*3.2. Computing the splines*

We need now to compute to $\widehat{x}_{\lambda,\tau_d}$ starting with $\mathbf{x}^{\tau_d} = (x(t))_{t\in\tau_d}^T$. This can be done via a theorem from Kimeldorf and Wahba (1971). We need the following compatibility assumptions between the sampling grid $\tau_d$ and the boundary conditions operator $B$:

**Assumption 1.** *The sampling grid $\tau_d = (t_l)_{l=1}^{|\tau_d|}$ is such that*

1. *sampling points are distinct in $[0,1]$ and $|\tau_d| \geq m-1$*
2. *the $m$ boundary conditions $B^j$ are linearly independent from the $|\tau_d|$ linear forms $h \mapsto h(t_l)$, for $l = 1, \ldots, |\tau_d|$ (defined on $\mathcal{H}^m$)*

Then $\widehat{x}_{\lambda,\tau_d}$ and $\mathbf{x}^{\tau_d} = (x(t))_{t\in\tau_d}^T$ are linked by the following result:

**Theorem 1** (Kimeldorf and Wahba (1971))**.** *Under Assumption (A1), the unique solution $\widehat{x}_{\lambda,\tau_d}$ to equation* (1) *is given by:*

$$\widehat{x}_{\lambda,\tau_d} = \mathcal{S}_{\lambda,\tau_d}\mathbf{x}^{\tau_d}, \tag{2}$$

*where $\mathcal{S}_{\lambda,\tau_d}$ is a full rank linear operator from $\mathbb{R}^{|\tau_d|}$ to $\mathcal{H}^m$ defined by:*

$$\mathcal{S}_{\lambda,\tau_d} = \omega^T M_0 + \eta^T M_1 \tag{3}$$

*with*

- $M_0 = \left(U(K_1 + \lambda I_d)^{-1}U^T\right)^{-1} U(K_1 + \lambda I_d)^{-1}$
- $M_1 = (K_1 + \lambda I_d)^{-1}\left(I_d - U^T M_0\right)$;
- $\{\omega_1, \ldots, \omega_m\}$ *is a basis of* $\mathbb{P}^{m-1}$, $\omega = (\omega_1, \ldots, \omega_m)^T$ *and* $U = (\omega_i(t))_{i=1,\ldots,m\ t\in\tau_d}$;
- $\eta = (k_1(t,.))_{t\in\tau_d}^T$ *and* $K_1 = (k_1(t,t'))_{t,t'\in\tau_d}$.

*3.3. No information loss*

The first important consequence of Theorem 1 is that building a model on $\widehat{X}_{\lambda,\tau_d}$ or on $\mathbf{X}^{\tau_d}$ leads to the same optimal predictive performances (to the same Bayes risk). This is formalized by the following corollary:

**Corollary 1.** *Under Assumption (A1), we have*

- *in the binary classification case:*

$$\inf_{\phi:\mathcal{H}^m\to\{-1,1\}} \mathbb{P}\left(\phi(\widehat{X}_{\lambda,\tau_d}) \neq Y\right) = \inf_{\phi:\mathbb{R}^{|\tau_d|}\to\{-1,1\}} \mathbb{P}\left(\phi(\mathbf{X}^{\tau_d}) \neq Y\right) \tag{4}$$

- *in the regression case:*

$$\inf_{\phi:\mathcal{H}^m\to\mathbb{R}} \mathbb{E}\left(\left[\phi\left(\widehat{X}_{\lambda,\tau_d}\right) - Y\right]^2\right) = \inf_{\phi:\mathbb{R}^{|\tau_d|}\to\mathbb{R}} \mathbb{E}\left(\left[\phi\left(\mathbf{X}^{\tau_d}\right) - Y\right]^2\right) \tag{5}$$

*3.4. Differentiation operator*

The second important consequence of Theorem 1 is that the inner product $\langle ., . \rangle_{\mathcal{H}^m}$ is equivalent to a specific inner product on $\mathbb{R}^{|\tau_d|}$ given in the following corollary:

**Corollary 2.** *Under Assumption (A1) and for any* $\mathbf{u}^{\tau_d} = (u(t))_{t\in\tau_d}^T$ *and* $\mathbf{v}^{\tau_d} = (v(t))_{t\in\tau_d}^T$ *in* $\mathbb{R}^{|\tau_d|}$,

$$\langle \widehat{u}_{\lambda,\tau_d}, \widehat{v}_{\lambda,\tau_d} \rangle_{\mathcal{H}^m} = (\mathbf{u}^{\tau_d})^T \mathbf{M}_{\lambda,\tau_d} \mathbf{v}^{\tau_d} \tag{6}$$

*where* $\mathbf{M}_{\lambda,\tau_d} = M_0^T W M_0 + M_1^T K_1 M_1$ *with* $W = (\langle w_i, w_j \rangle_0)_{i,j=1,\ldots,m}$. *The matrix* $\mathbf{M}_{\lambda,\tau_d}$ *is symmetric and positive definite and defines an inner product on* $\mathbb{R}^{|\tau_d|}$.

The corollary is a direct consequence of equations (2) and (3).

In practice, the corollary means that the euclidean space $\left(\mathbb{R}^{|\tau_d|}, \langle ., . \rangle_{\mathbf{M}_{\lambda,\tau_d}}\right)$ is isomorphic to $\left(\mathcal{I}_{\lambda,\tau_d}, \langle ., . \rangle_{\mathcal{H}^m}\right)$, where $\mathcal{I}_{\lambda,\tau_d}$ is the image of $\mathbb{R}^{|\tau_d|}$ by $\mathcal{S}_{\lambda,\tau_d}$. As a consequence, one can use the Hilbert structure of $\mathcal{H}^m$ directly in $\mathbb{R}^{|\tau_d|}$ via $\mathbf{M}_{\lambda,\tau_d}$: as the inner product of $\mathcal{H}^m$ is defined on the order $m$ derivatives of the functions, this corresponds to using those derivatives instead of the original functions.

More precisely, let $\mathbf{Q}_{\lambda,\tau_d}$ be the transpose of the Cholesky triangle of $\mathbf{M}_{\lambda,\tau_d}$ (given by the Cholesky decomposition $\mathbf{Q}_{\lambda,\tau_d}^T \mathbf{Q}_{\lambda,\tau_d} = \mathbf{M}_{\lambda,\tau_d}$). Corollary 2 shows that $\mathbf{Q}_{\lambda,\tau_d}$ acts as an approximate differentiation operation on sampled functions.

Let us indeed consider an estimation method for multivariate inputs based only on inner products or norms (that are directly derived from the inner products), such as, e.g., Kernel Ridge Regression Saunders et al. (1998); Shawe-Taylor and Cristianini (2004). In this latter case, if a Gaussian kernel is used, the regression function has the following form:

$$u \mapsto \sum_{i=1}^{n} T_i \alpha_i e^{-\gamma \|U_i - u\|_{\mathbb{R}^p}^2} \tag{7}$$

where $(U_i, T_i)_{1 \leq i \leq n}$ are learning examples in $\mathbb{R}^p \times \{-1, 1\}$ and the $\alpha_i$ are non negative real values obtained by solving a quadratic programming problem and $\gamma$ is a parameter of the method. Then, if we use Kernel Ridge Regression on the training set $\{(\mathbf{Q}_{\lambda,\tau_d} \mathbf{X}_i^{\tau_d}, Y_i)\}_{i=1}^n$ (rather than the original training set $\{(\mathbf{X}_i^{\tau_d}, Y_i)\}_{i=1}^n$), it will work on the norm in $L^2$ of the derivatives of order $m$ of the spline estimates of the $X_i$ (up to the boundary conditions). More precisely, the regression function will have the following form:

$$\begin{aligned}
\mathbf{x}^{\tau_d} &\mapsto \sum_{i=1}^{n} Y_i \alpha_i e^{-\gamma \left\| \mathbf{Q}_{\lambda,\tau_d} \mathbf{X}_i^{\tau_d} - \mathbf{Q}_{\lambda,\tau_d} \mathbf{x}^{\tau_d} \right\|_{\mathbb{R}^{|\tau_d|}}^2} \\
&\mapsto \sum_{i=1}^{n} Y_i \alpha_i e^{-\gamma \left\| D^m \widehat{X_i}_{\lambda,\tau_d} - D^m \widehat{x}_{\lambda,\tau_d} \right\|_{L^2}^2} \\
&\quad \times e^{-\gamma \sum_{j=1}^{m} \left( B^j \widehat{X_i}_{\lambda,\tau_d} - B^j \widehat{x}_{\lambda,\tau_d} \right)^2}
\end{aligned}$$

In other words, up to the boundary conditions, an estimation method based solely on inner products, or on norms derived from these inner products,

can be given modified inputs that will make it work on an estimation of the derivatives of the observed functions.

**Remark 1.** As shown in Corollary 1 in the previous section, building a model on $\mathbf{X}^{\tau_d}$ or on $\widehat{X}_{\lambda,\tau_d}$ leads to the same optimal predictive performances. In addition, it is obvious that given any one-to-one mapping $f$ from $\mathbb{R}^{|\tau_d|}$ to itself, building a model on $f(\mathbf{X}^{\tau_d})$ gives also the same optimal performances than building a model on $\mathbf{X}^{\tau_d}$. Then as $\mathbf{Q}_{\lambda,\tau_d}$ is invertible, the optimal predictive performances achievable with $\mathbf{Q}_{\lambda,\tau_d}\mathbf{X}^{\tau_d}$ are equal to the optimal performances achievable with $\mathbf{X}^{\tau_d}$ or with $\widehat{X}_{\lambda,\tau_d}$.

In practice however, the actual preprocessing of the data can have a strong influence on the obtained performances, as will be illustrated in Section 6. The goal of the theoretical analysis of the present section is to guarantee that no systematic loss can be observed as a consequence of the proposed functional preprocessing scheme.

## 4. Approximation results

The previous section showed that working on $\mathbf{X}^{\tau_d}$, $\mathbf{Q}_{\lambda,\tau_d}\mathbf{X}^{\tau_d}$ or $\widehat{X}_{\lambda,\tau_d}$ makes no difference in terms of optimal predictive performances. The present section addresses the effects of sampling: asymptotically, the optimal predictive performances obtained on $\widehat{X}_{\lambda,\tau_d}$ converge to the optimal performances achievable on the original and unobserved functional variable $X$.

### *4.1. Spline approximation*

From the sampled random function $\mathbf{X}^{\tau_d} = (X(t_1), \ldots, X(t_{|\tau_d|}))$, we can build an estimate, $\widehat{X}_{\lambda,\tau_d}$, of $X$. To ensure consistency, we must guarantee that $\widehat{X}_{\lambda,\tau_d}$ converges to $X$. In the case of a deterministic function $x$, this problem has been studied in numerous papers, such as Craven and Wahba (1978); Ragozin (1983); Cox (1984); Utreras (1988); Wahba (1990) (among others). Here we recall one of the results which is particularly well adapted to our context.

Obviously, the sampling grid must behave correctly, whereas the information contained in $\mathbf{X}^{\tau_d}$ will not be sufficient to recover $X$. We need also the regularization parameter $\lambda$ to depend on $\tau_d$. Following Ragozin (1983), a sampling grid $\tau_d$ is characterized by two quantities:

$$\begin{aligned} \overline{\Delta}_{\tau_d} &= \max\{t_1, t_2 - t_1, \ldots, 1 - t_{|\tau_d|}\} \\ \underline{\Delta}_{\tau_d} &= \min_{1 \leq i < |\tau_d|} \{t_{i+1} - t_i\}. \end{aligned} \tag{8}$$

One way to control the distance between $X$ and $\widehat{X}_{\lambda,\tau_d}$ is to bound the ratio $\overline{\Delta}_{\tau_d}/\underline{\Delta}_{\tau_d}$ so as to ensure quasi-uniformity of the sampling grid.

More precisely, we will use the following assumption:

**Assumption 2.** *There is $R$ such that $\overline{\Delta}_{\tau_d}/\underline{\Delta}_{\tau_d} \leq R$ for all $d$.*

Then we have:

**Theorem 2** (Ragozin (1983))**.** *Under Assumptions (A1) and (A2), there are two constants $A_{R,m}$ and $B_{R,m}$ depending only on $R$ and $m$, such that for any $x \in \mathcal{H}^m$ and any positive $\lambda$:*

$$\|\widehat{x}_{\lambda,\tau_d} - x\|_{L^2}^2 \leq \left(A_{R,m}\lambda + B_{R,m}\frac{1}{|\tau_d|^{2m}}\right)\|D^m x\|_{L^2}^2 .$$

This result is a rephrasing of Corollary 4.16 from Ragozin (1983) which is itself a direct consequence of Theorem 4.10 from the same paper.

Convergence of $\widehat{x}_{\lambda,\tau_d}$ to $x$ is then obtained by the following simple assumptions:

**Assumption 3.** *The series of sampling points $\tau_d$ and the series of regularization parameters, $\lambda$, depending on $\tau_d$ and denoted by $(\lambda_d)_{d\geq 1}$, are such that $\lim_{d\to+\infty}|\tau_d| = +\infty$ and $\lim_{d\to+\infty}\lambda_d = 0$.*

*4.2. Conditional expectation approximation*

The next step consists in relating the optimal predictive performances for the regression and the classification problem $(X, Y)$ to the performances associated to $(\widehat{X}_{\lambda_d,\tau_d}, Y)$ when $d$ goes to infinity, i.e., relating $L^*$ to

1. *binary classification case*:

$$L_d^* = \inf_{\phi:\mathcal{H}^m\to\{-1,1\}} \mathbb{P}\left(\phi(\widehat{X}_{\lambda_d,\tau_d}) \neq Y\right),$$

2. *regression case*:

$$L_d^* = \inf_{\phi:\mathcal{H}^m\to\mathbb{R}} \mathbb{E}\left([\phi(\widehat{X}_{\lambda_d,\tau_d}) - Y]^2\right)$$

Two sets of assumptions will be investigated to provide the convergence of the Bayes risk $L_d^*$ to $L^*$:

**Assumption 4. Either**

*(A4a)* $\mathbb{E}\left(\|D^m X\|_{L^2}^2\right)$ *is finite and* $Y \in \{-1, 1\}$,
**or**

*(A4b)* $\tau_d \subset \tau_{d+1}$ *and* $\mathbb{E}(Y^2)$ *is finite.*

The first assumption (A4a) requires an additional smoothing property for the predictor functional variable $X$ and is only valid for a binary classification problem whereas the second assumption (A4a) requires an additional property for the sampling point series: they have to be growing sets.

Theorem 2 then leads to the following corollary:

**Corollary 3.** *Under Assumptions (A1)-(A4), we have:*

$$\lim_{d \to +\infty} L_d^* = L^*.$$

## 5. General consistent functional classifiers and regression functions

### *5.1. Definition of classifiers and regression functions on derivatives*

Let us now consider any consistent classification or regression scheme for standard multivariate data based either on the inner product or on the Euclidean distance between observations. Examples of such classifiers are Support Vector Machine Steinwart (2002), the kernel classification rule Devroye and Krzyżak (1989) and $k$-nearest neighbors Devroye and Györfi (1985); Zhao (1987) to name a few. In the same way, multilayer perceptrons Lugosi and Zeger (1990), kernel estimates Devroye and Krzyżak (1989) and $k$-nearest neighbors regression Devroye et al. (1994) are consistent regression estimators. Additional examples of consistent estimators in classification and regression can be found in Devroye et al. (1996); Györfi et al. (2002).

We denote $\psi_{\mathcal{D}}$ the estimator constructed by the chosen scheme using a dataset $\mathcal{D} = \{(U_i, T_i)_{1 \leq i \leq n}\}$, where the $(U_i, T_i)_{1 \leq i \leq n}$ are $n$ independent copies of a pair of random variables $(U, T)$ with values in $\mathbb{R}^p \times \{-1, 1\}$ (classification) or $\mathbb{R}^p \times \mathbb{R}$ (regression).

The proposed functional scheme consists in choosing the estimator $\phi_{n,\tau_d}$ as $\psi_{\mathcal{E}_{n,\tau_d}}$ with the dataset $\mathcal{E}_{n,\tau_d}$ defined by:

$$\mathcal{E}_{n,\tau_d} = \{(\mathbf{Q}_{\lambda_d,\tau_d} \mathbf{X}_i^{\tau_d}, Y_i)_{1 \leq i \leq n}\}$$

As pointed out in Section 3.4, the linear transformation $\mathbf{Q}_{\lambda_d,\tau_d}$ is an approximate multivariate differentiation operator: up to the boundary conditions, an estimator based on $\mathbf{Q}_{\lambda_d,\tau_d}\mathbf{X}^{\tau_d}$ is working on the $m$-th derivative of $\widehat{X}_{\lambda_d,\tau_d}$.

In more algorithmic terms, the estimator is obtained as follows:

1. choose an appropriate value for $\lambda_d$
2. compute $\mathbf{M}_{\lambda_d,\tau_d}$ using Theorem 1 and Corollary 2;
3. compute the Cholesky decomposition of $\mathbf{M}_{\lambda_d,\tau_d}$ and the transpose of the Cholesky triangle, $\mathbf{Q}_{\lambda_d,\tau_d}$ (such that $\mathbf{Q}^T_{\lambda_d,\tau_d}\mathbf{Q}_{\lambda_d,\tau_d} = \mathbf{M}_{\lambda_d,\tau_d}$);
4. compute $\mathbf{Q}_{\lambda_d,\tau_d}\mathbf{X}_i^{\tau_d}$ to obtain the transformed dataset $\mathcal{E}_{n,\tau_d}$;
5. build a classifier/regression function $\psi_{\mathcal{E}_{n,\tau_d}}$ with a multivariate method in $\mathbb{R}^{|\tau_d|}$ applied to the dataset $\mathcal{E}_{n,\tau_d}$;
6. associate to a new sampled function $\mathbf{X}_{n+1}^{\tau_d}$ the prediction $\psi_{\mathcal{E}_{n,\tau_d}}(\mathbf{Q}_{\lambda,\tau_d}\mathbf{X}_{n+1}^{\tau_d})$.

Figure 5.1 illustrates the way the method performs: instead of relying on an approximation of the function and then on the derivation preprocessing of this estimates, it directly uses an equivalent metric by applying the $\mathbf{Q}_{\lambda_d,\tau_d}$ matrix to the sampled function. The consistency result proved in Theorem 3 shows that, combined with any consistent multidimensional learning algorithm, this method is (asymptotically) equivalent to using the original function drawn at the top left side of Figure 5.1.

On a practical point of view, Wahba (1990) demonstrates that cross validated estimates of $\lambda$ achieve suitable convergence rates. Hence, steps 1 and 2 can be computed simultaneously by minimizing the total cross validated error for all the observations, given by

$$\sum_{i=1}^{n} \frac{1}{|\tau_d|} \sum_{t\in\tau_d} \frac{\left(x_i(t) - \widehat{x}_{i\lambda,\tau_d}(t)\right)^2}{\left(1 - A_{tt}(\lambda)\right)^2},$$

where $A$ is a $|\tau_d| \times |\tau_d|$ matrix called the *influence matrix* (see Wahba (1990)), over a finite number of $\lambda$ values.

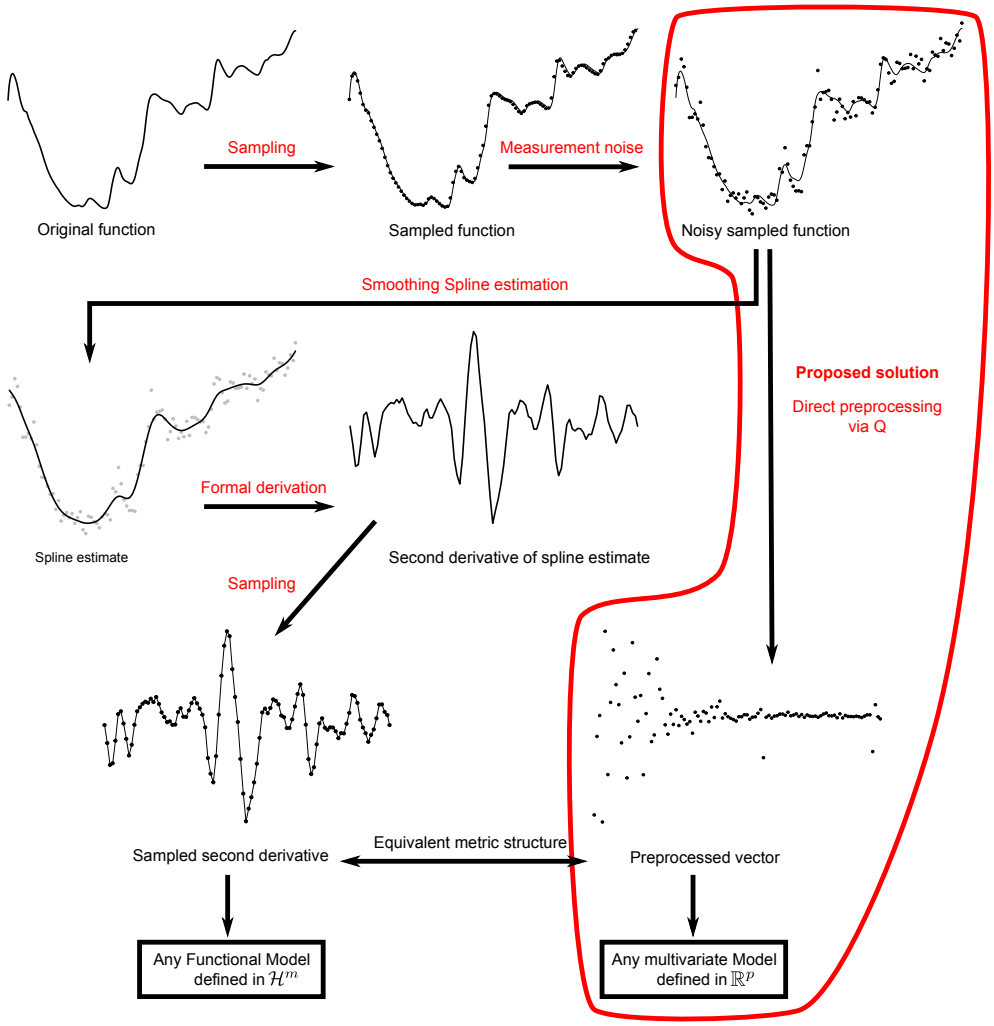


Figure 1: Method scheme and its equivalence to the usual approach for using derivatives in learning algorithms.

*5.2. Consistency result*

Corollary 1 and Corollary 3 guarantee that the estimator proposed in the previous section is consistent:

**Theorem 3.** *Under assumptions (A1)-(A4), the series of classifiers/regression functions* $(\phi_{n,\tau_d})_{n,d}$ *is consistent:*

$$\lim_{d\to+\infty}\lim_{n\to+\infty}\mathbb{E}\left(L\phi_{n,\tau_d}\right)=L^*$$

*5.3. Discussion*

While Theorem 3 is very general, it could be easily extended to cover special cases such as additional hypothesis needed by the estimation scheme or to provide data based parameter selections. We discuss briefly those issues in the present section.

It should first be noted that most estimation schemes, $\psi_{\mathcal{D}}$, depend on parameters that should fulfill some assumptions for the scheme to be consistent. For instance, in the Kernel Ridge Regression method in $\mathbb{R}^p$, with Gaussian kernel, $\psi_{\mathcal{D}}$ has the form given in Equation (7) where the $(\alpha_i)$ are the solutions of

$$\arg\min_{\alpha\in\mathbb{R}^n}\sum_{i=1}^{n}\left(T_i-\sum_{j=1}^{n}T_j\alpha_j e^{-\gamma\|U_i-U_j\|^2_{\mathbb{R}^p}}\right)^2+$$
$$\delta_n\sum_{i,j=1}^{n}T_iT_j\alpha_i\alpha_j e^{-\gamma\|U_i-U_j\|^2_{\mathbb{R}^p}}.$$

The method thus depends on the parameter of the Gaussian kernel, $\gamma$ and of the regularization parameter $\delta_n$. This method is known to be consistent if (see Theorem 9.1 of Steinwart and Christmann (2008)):

$$\delta_n\xrightarrow{n\to+\infty}0\qquad\text{and}\qquad n\delta_n^4\xrightarrow{n\to+\infty}+\infty.$$

Additional conditions of this form can obviously be directly integrated in Theorem 3 to obtain consistency results specific to the corresponding algorithms.

Moreover, practitioners generally rely on data based selection of the parameters of the estimation scheme $\psi_{\mathcal{D}}$ via a validation method: for instance, rather than setting $\delta_n$ to e.g., $n^{-5}$ for $n$ observations (a choice which is compatible with theoretical constraints on $\delta_n$), one chooses the value of $\delta_n$ that

optimizes an estimation of the performances of the regression function obtained on an independent data set (or via a re-sampling approach).

In addition to the parameters of the estimation scheme, functional data raise the question of the convenient order of the derivative, $m$, and of the sampling grid optimality. In practical applications, the number of available sampling points can be unnecessarily large (see Biau et al. (2005) for an example with more than 8 000 sampling points). The preprocessing performed by $\mathbf{Q}_{\lambda_d,\tau_d}$ do not change the dimensionality of the data which means that overfitting can be observed in practice when the number of sampling points is large compared to the number of functions. Moreover, processing very high dimensional vectors is time consuming. It is there quite interesting in practice to use a down-sampled version of the original grid.

To select the parameters of $\psi_D$, the order of the derivative and/or the down-sampled grid, a validation strategy, based on splitting the dataset into training and validation sets, could be used. A simple adaptation of the idea of Berlinet et al. (2008); Biau et al. (2005); Laloë (2008); Rossi and Villa (2006) shows that a penalized validation method can be used to choose any combination of those parameters consistently. According to those papers, the condition for the consistency of the validation strategy would simply relate the shatter coefficients of the set of classifiers in $\mathbb{R}^d$ to the penalization parameter of the validation. Once again, this type of results is a rather direct extension of Theorem 3.

## 6. Applications

In this section, we show that the proposed approach works as expected on real world spectrometric examples: for some applications, the use of derivatives leads to more accurate models than the direct processing of the spectra (see e.g. Rossi et al. (2005); Rossi and Villa (2006) for other examples of such a behavior based on ad hoc estimators of the spectra derivatives). It should be noted that the purpose of this section is only to illustrate the behavior of the proposed method on finite datasets. The theoretical results of the present paper show that all consistent schemes have asymptotically identical performances, and therefore that using derivatives is asymptotically useless. On a finite dataset however, preprocessing can have strong influence on the predictive performances, as will be illustrated in the present section. In addition, schemes that are not universally consistent, e.g., linear models, can lead to excellent predictive performances on finite datasets; such models are

therefore included in the present section despite the fact the theory does not apply to them.

### *6.1. Methodology*

The methodology followed for the two illustrative datasets is roughly the same:

1. the dataset is randomly split into a training set on which the model is estimated and a test set on which performances are computed. The split is repeated several times. The Tecator dataset (Section 6.2) is rather small (240 spectra) and exhibits a rather large variability in predictive performances between different random splits. We have therefore used 250 random splits. For the Yellow-berry dataset (Section 6.3), we used only 50 splits as the relative variability in performances is far less important.

2. $\lambda$ is chosen by a global leave-one-out strategy on the spectra contained in training set (as suggested in Section 5.1). More precisely, a leave-one-out estimate of the reconstruction error of the spline approximation of each training spectrum is computed for a finite set of candidate values for $\lambda$. Then a common $\lambda$ is chosen by minimizing the average over the training spectra of the leave-one-out reconstruction errors. This choice is relevant as cross validation estimates of $\lambda$ are known to have favorable theoretical properties (see Craven and Wahba (1978); Utreras (1981) among others).

3. for regression problems, a Kernel Ridge Regression (KRR) Saunders et al. (1998); Shawe-Taylor and Cristianini (2004) is then performed to estimate the regression function; this method is consistent when used with a Gaussian kernel under additional conditions on the parameters (see Theorem 9.1 of Steinwart and Christmann (2008)); as already explained, in the applications, Kernel Ridge Regression is performed both with a Gaussian kernel and with a linear kernel (in that last case, the model is essentially a ridge regression model). Parameters of the models (a regularization parameter, $\delta_n$, in all cases and a kernel parameter, $\gamma$ for Gaussian kernels) are chosen by a grid search that minimizes a validation based estimate of the performances of the model (on the training set). A leave-one-out solution has been chosen: in Kernel Ridge Regression, the leave-one-out estimate of the performances of the model is

obtained as a by-product of the estimation process, without additional computation cost, see e.g. Cawley and Talbot (2004).
Additionally, for a sake of comparison with a more traditional approach in FDA, Kernel Ridge Regression is compared with a nonparametric kernel estimate for the Tecator dataset (Section 6.2.1). Nonparametric kernel estimate is the first nonparametric approach introduced in Functional Data Analysis Ferraty and Vieu (2006) and can thus be seen as a basis for comparison in the context of regression with functional predictors. For this method, the same methodology as with Kernel Ridge Regression was used: the parameter of the model (i.e., the bandwidth) was selected on a grid search minimizing a cross-validation estimate of the performances of the model. In this case, a 4-fold cross validation estimate was used instead of a leave-one-out estimate to avoid a large computational cost.

4. for the classification problem, a Support Vector Machine (SVM) is used Shawe-Taylor and Cristianini (2004). As KRR, SVM are consistent when used with a Gaussian kernel Steinwart (2002). We also use a SVM with a linear kernel as this is quite adapted for classification in high dimensional spaces associated to sampled function data. We also use a K-nearest neighbor model (KNN) for reference. Parameters of the models (a regularization parameter for both SVM, a kernel parameter, $\gamma$ for Gaussian kernels and number of neighbors K for KNN) are chosen by a grid search that minimizes a validation based estimate of the classification error: we use a 4-fold cross-validation to get this estimate.

5. We evaluate the models obtained for each random split on the test set. We report the mean and the standard deviation of the performance index (classification error and mean squared error, respectively) and assess the significance of differences between the reported figures via paired Student tests (with level 1%).

6. Finally, we compare models estimated on the raw spectra and on spectra transformed via the $\mathbf{Q}_{\lambda_d,\tau_d}$ matrix for $m = 1$ (first derivative) and $m = 2$ (second derivative). For both values of $m$, we used the most classical boundary conditions ($x(0) = 0$ and $Dx(0) = 0$). Depending of the problem, other boundary conditions could be investigated but this is outside the scope of the present paper (see Besse and Ramsay (1986); Heckman and Ramsay (2000) for discussion on this subject). For the

Tecator problem, we also compare these approaches with models estimated on first and second derivatives based on interpolating splines (i.e. with $\lambda = 0$) and on first and second derivatives estimated by finite differences.

Note that the kind of preprocessing used has almost no impact on the computation time. In general, selecting the parameters of the model with leave-one-out or cross-validation will use significantly more computing power than constructing the splines and calculating their derivatives. For instance, computing the optimal $\lambda$ with the approach described above takes less than 0.1 second for the Tecator dataset on a standard PC using our R implementation which is negligible compared to the several minutes used to select the optimal parameters of the models used on the prepocessed data.

### *6.2. Tecator dataset*

The first studied dataset is the standard Tecator dataset Thodberg (1996) [1]. It consists in spectrometric data from the food industry. Each of the 240 observations is the near infrared absorbance spectrum of a meat sample recorded on a Tecator Infratec Food and Feed Analyzer. Each spectrum is sampled at 100 wavelengths uniformly spaced in the range 850–1050 nm. The composition of each meat sample is determined by analytic chemistry and percentages of moisture, fat and protein are associated this way to each spectrum.

The Tecator dataset is a widely used benchmark in Functional Data Analysis, hence the motivation for its use for illustrative purposes. More precisely, in Section 6.2.1, we address the original regression problem by predicting the percentage of fat content from the spectra with various regression method and various estimates of the derivative preprocessing: this analysis shows that both the method and the use of derivative have a strong effect on the performances whereas the way the derivatives are estimated has almost no effect. Additionally, in Section 6.2.2, we apply a noise (with various variances) to the original spectra in order to study the influence of smoothing in the case of noisy predictors: this section shows the relevance of the use of a smoothing spline approach when the data are noisy. Finally, Section 6.2.3 deals with a classification problem derived from the original Tecator problem

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

(in the same way as what was done in Ferraty and Vieu (2003)): conclusions of this section are similar to the ones of the regression study.

*6.2.1. Fat content prediction*

As explained above, we first address the regression problem that consists in predicting the fat content of peaces of meat from the Tecator dataset. The parameters of the model are optimized with a grid search using the leave-one-out estimate of the predictive performances (both models use a regularization parameter, with an additional width parameter in the Gaussian kernel case). The original data set is split randomly into 160 spectra for learning and 80 spectra for testing. As shown in the result Table 1, the data exhibit a rather large variability; we use therefore 250 random split to assess the differences between the different approaches.

The performance indexes are the mean squared error (M.S.E.) and the $R^2$.[2] As a reference, the target variable (fat) has a variance equal to 14.36. Results are summarized in Table 1.

The first conclusion is that the method itself has a strong effect on the performances of the prediction: for this application, a linear method is not appropriate (mean squared errors are much greater for linear methods than for the kernel ridge regression used with a Gaussian kernel) and the non-parametric kernel estimate gives worse performances than the kernel ridge regression (indeed, they are about 10 times worse). Nevertheless, for non-parametric approaches (Gaussian KKR and NKE), the use of derivatives has also a strong impact on the performances: for kernel ridge regression, e.g., preprocessing by estimating the first order derivative leads to a strong decrease of the mean squared error.

Differences between the average MSEs are not always significant, but we can nevertheless rank the methods in increasing order of modeling error (using notations explained in Table 1) for Gaussian kernel ridge regression:

$$\mathrm{FD1} \leq \mathrm{IS1} \leq \mathrm{S1} < \mathrm{DF2} \leq \mathrm{SS2} < \mathrm{IS2} < \mathrm{O}$$

where $<$ corresponds to a significant difference (for a paired Student test with level 1%) and $\leq$ to a non significant one. In this case, the data are very smooth and thus the use of smoothing splines instead of a finite differences

[2] $R^2 = 1 - \frac{\mathrm{M.S.E}}{\mathrm{Var}(y)}$ where $\mathrm{Var}(y)$ is the (empirical) variance of the target variable on the test set.

| Method | Data | Average M.S.E. and SD | Average $R^2$ |
|---|---|---|---|
| KRR Linear | O | 8.69 (4.47) | 95.7% |
| | S1 | 8.09 (3.85) | 96.1% |
| | IS1 | 8.09 (3.85) | 96.1% |
| | FD1 | 8.27 (4.17) | 96.0% |
| | S2 | 9.64 (4.98) | 95.3% |
| | IS2 | 9.87 (5.84) | 95.2% |
| | FD2 | 8.45 (4.18) | 95.9% |
| KRR Gaussian | O | 5.02 (11.47) | 97.6% |
| | S1 | 0.485 (0.385) | 99.8% |
| | IS1 | 0.485 (0.385) | 99.8% |
| | FD1 | **0.484** (0.387) | **99.8%** |
| | S2 | 0.584 (0.303) | 99.7% |
| | IS2 | 0.586 (0.303) | 99.7% |
| | FD2 | 0.569 (0.281) | 99.7% |
| NKE | O | 73.1 (16.5) | 64.2% |
| | S1 | 4.59 (1.09) | 97.7% |
| | IS1 | 4.59 (1.09) | 97.7% |
| | FD1 | 4.59 (1.09) | 97.7% |
| | S2 | 3.75 (1.22) | 98.2% |
| | IS2 | 3.75 (1.22) | 98.2% |
| | FD2 | 3.67 (1.18) | 98.2% |

Table 1: Summary of the performances of the chosen models on the test set (fat Tecator regression problem) when using either a kernel ridge regression (KRR) with linear kernel or with Gaussian kernel or when using a nonparametric kernel estimate (NKE) with various inputs: O (original data), S1 (smoothing splines with order 1 derivatives), IS1 (interpolating splines with order 1 derivatives), FD1 (order 1 derivatives estimated by finite differences) and S2, IS2 and FD2 (the same as previously with order 2 derivatives).

approximation does not have a significant impact on the predictions. However, in this case, the roughest approach, consisting in the estimation of the derivatives by finite differences, gives the best performances.

*6.2.2. Noisy spectra*

This section studies the situation in which functional data observations are corrupted by noise. This is done by adding a noise to each spectrum of the Tecator dataset. More precisely, each spectrum has been corrupted by

$$X_i^b(t) = X_i(t) + \epsilon_{it} \tag{9}$$

where $(\epsilon_{it})$ are i.i.d. Gaussian variables with standard deviation equal to either 0.01 (small noise) or to 0.2 (large noise). 10 observations of the data generated this way are given in Figure 2.

The same methodology as for the non noisy data has been applied to $(X_i^b)$ to predict the fat content. The experiments have been restricted to the use of kernel ridge regression with a Gaussian kernel (according to the nonlinearity of the problem shown in the previous section). Results are summarized in Table 2 and Figure 3.

In addition, the results can be ranked this way:

**Noise with sd equal to 0.01**

$$\text{S2} < \text{S1} < \text{IS1} \leq \text{O} < \text{FD1} < \text{IS2} \leq \text{FD2}$$

**Noise with sd equal to 0.2**

$$\text{S1} < \text{O} < \text{S2} < \text{FD1} < \text{IS1} < \text{IS2} \leq \text{FD2}$$

where $<$ corresponds to a significant difference (for a paired Student test with level 1%).

The first conclusion of these experiments is that, even though the derivatives are the relevant predictors, their performances are strongly affected by the noise (compared to the ones of the original data: note that the average M.S.E. reported in Table 1 are more 10 times lower that the best ones from Table 2 and that, in the best cases, $R^2$ is slightly greater than 50% for the most noisy dataset). In particular, using interpolating splines or finite difference derivatives leads to highly deteriorated performances. In this situation, the approach proposed in the paper is particularly useful and helps to keep

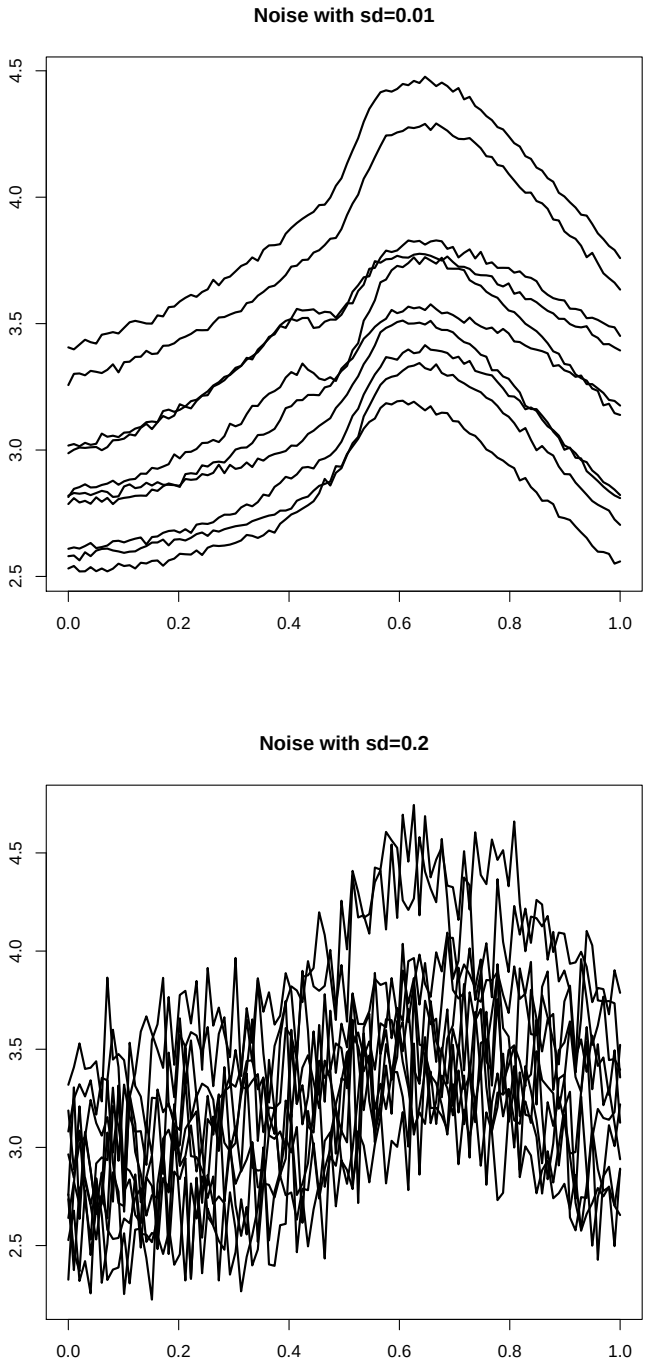


Figure 2: 10 observations of the noisy data generated from the Tecator spectra as in Equation 9

| Noise | Data | Average M.S.E. and SD | Average $R^2$ |
|---|---|---|---|
| sd = 0.01 | O | 13.3 (13.5) | 93.5% |
| | S1 | 7.45 (1.5) | 96.4% |
| | IS1 | 12.72 (2.2) | 93.8% |
| | FD1 | 20.03 (2.8) | 90.3% |
| | S2 | **6.83** (1.4) | **96.7%** |
| | IS2 | 31.23 (5.9) | 84.9% |
| | FD2 | 31.10 (5.9) | 84.9% |
| sd = 0.2 | O | 87.9 (13.9) | 57.4% |
| | S1 | **85.0** (12.5) | **58.8**% |
| | IS1 | 210.1 (36.1) | -1.9% |
| | FD1 | 209.1 (33.0) | -1.4% |
| | S2 | 95.9 (12.8) | 53.5% |
| | IS2 | 213.7 (33.1) | -3.6% |
| | FD2 | 235.1 (222.7) | -14.0% |

Table 2: Summary of the performances of the chosen models on the test set (fat Tecator regression problem) with noisy spectra.

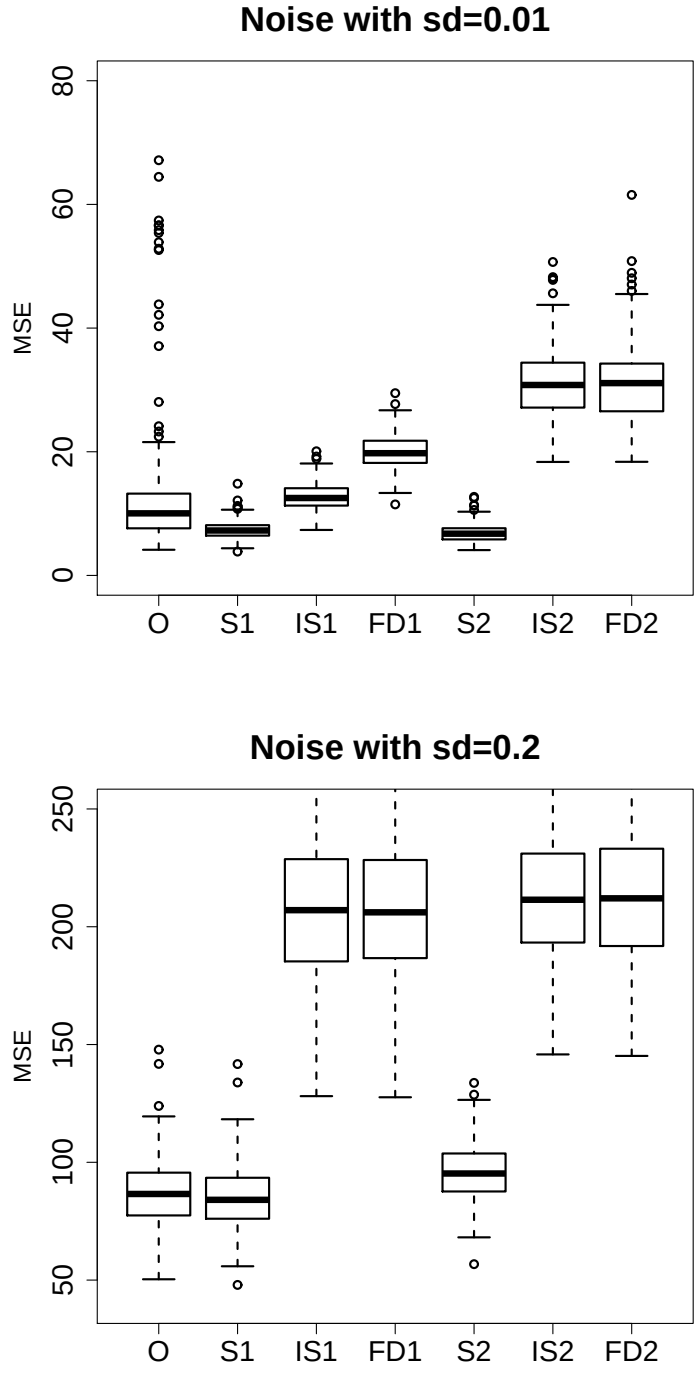


Figure 3: Mean squared errors boxplot for the noisy fat Tecator regression problem with Gaussian kernel (the worst test samples for IS and FD have been removed for a sake of clarity)

better performances than with the original data. Indeed, the differences of the smoothing splines approach with the original data is still significant (for both derivatives in the "small noise" case and for the first order derivative in the "high noise" case), even though, the most noisy the data are, the most difficult it is to estimate the derivatives in an accurate way. That is, except for smoothing spline derivatives, the estimation of the derivatives for the most noisy dataset is so bad that it leads to negative $R^2$ when used in the regression task.

*6.2.3. Fat content classification*

In this section, the fat content regression problem is transformed into a classification problem. To avoid imbalance in class sizes, the median value of the fat in the dataset is used as the splitting criterion: the first class consists in 119 samples with strictly less than 13.5 % of fat, while the second class contains the other 121 samples with a fat content equal or higher than 13.5 %.

As in previous sections, the analysis is conducted on 250 random splits of the dataset into 160 learning spectra and 80 test spectra. We used stratified sampling: the test set contains 40 examples from each class. The 4 fold cross-validation used to select the parameters of the models on the learning set is also stratified with roughly 20 examples of each class in each fold.

The performance index is the mis-classification rate (MCR) on the test set, reported in percentage and averaged over the 250 random splits. Results are summarized in Table 3. As in the previous sections, both the model and the preprocessing have some influence on the results. In particular, using derivatives always improves the classification accuracy while the actual method used to compute those derivatives has no particular influence on the results. Additionally, using interpolation splines leads, in this particular problem, to results that are exactly identical to the ones obtained with the smoothing splines: they are not reported in Table 3.

More precisely, for the three models (linear SVM, Gaussian SVM and KNN), differences in mis-classification rates between the smoothing spline preprocessing and the finite differences calculation is never significant, according to a Student test with level 1 %. Additionally while the actual average mis-classification rates might seem quite different, the large variability of the results (shown by the standard deviations) leads to significant differences only for the most obvious cases. In particular, SVM models using derivatives (of order one or two) are indistinguishable one from another using a Student

| Method | Data | Average MCR | SD of MCR |
|---|---|---|---|
| Linear SVM | O | 1.41 | 1.55 |
| | S1 | **0.73** | 1.15 |
| | FD1 | 0.74 | 1.15 |
| | S2 | 0.94 | 1.27 |
| | FD2 | 0.92 | 1.23 |
| Gaussian SVM | O | 3.39 | 2.57 |
| | S1 | 0.97 | 1.41 |
| | FD1 | 0.98 | 1.42 |
| | S2 | 0.99 | 2.00 |
| | FD2 | 0.97 | 1.27 |
| KNN | O | 22.0 | 5.02 |
| | S1 | 6.67 | 2.55 |
| | FD1 | 6.57 | 2.55 |
| | S2 | 1.93 | 1.65 |
| | FD2 | 1.93 | 1.63 |

Table 3: Summary of the performances of the chosen models on the test set (Tecator fat classification problem). See Table 1 for notations. MCR stands for mis-classification rate, SD for standard deviation.

test with level 1 %: all methods with less than 1 % of mean mis-classification rate perform essentially identically. Other differences are significant: for instance the linear SVM used on raw data performs significantly worse than any SVM model used on derivatives.

It should be noted that the classification task studied in the present section is obviously simpler than the regression task from which it is derived. This explains the very good predictive performances obtained by simple models such as a linear SVM, especially with the proper preprocessing.

### *6.3. Yellow-berry dataset*

The goal of the last experiment is to predict the presence of yellow-berry in durum wheat (*Triticum durum*) kernels via a near infrared spectral analysis (see Figure 4). Yellow-berry is a defect of the durum wheat seeds that reduces the quality of the flour produced from affected wheat. The traditional way to assess the occurrence of yellow-berry is by visual analysis of a sample of the seed stock. In the current application, a quality measure related to the occurrence of yellow-berry is predicted from the spectrum of the seed.

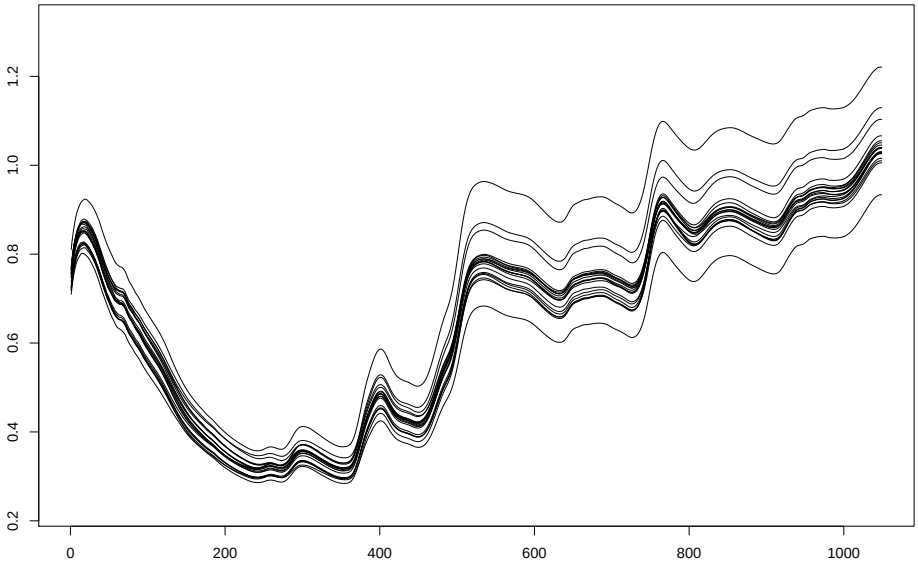


Figure 4: 20 observations of NIR spectra of durum wheat

The dataset consists in 953 spectra sampled at 1049 wavelengths uniformly spaced in the range 400–2498 nm. The dataset is split randomly into 600 learning spectra and 353 test spectra. Comparatively to the Tecator dataset, the variability of the results is smaller in the present case. We used therefore 50 random splits rather than 250 in the previous section.

The regression models were build via a Kernel Ridge Regression approach using a linear kernel and a Gaussian kernel. In both cases, the regularization parameter of the model is optimized by a leave-one-out approach. In addition, the width parameter of the Gaussian kernel is optimized via the same procedure at the same time.

The performance index is the mean squared error (M.S.E.). As a reference, the target variable has a variance of 0.508. Results are summarized in Table 4 and Figure 5.

| Kernel and Data | Average M.S.E. | Standard deviation | Average $R^2$ |
|---|---|---|---|
| Linear-O | 0.122 | $8.77\,10^{-3}$ | 76.1% |
| Linear-S1 | 0.138 | $9.53\,10^{-3}$ | 73.0% |
| Linear-S2 | 0.122 | $8.41\,10^{-3}$ | 76.1% |
| Gaussian-O | 0.110 | $20.2\,10^{-3}$ | 78.5% |
| Gaussian-S1 | 0.0978 | $7.92\,10^{-3}$ | 80.9% |
| Gaussian-S2 | 0.0944 | $8.35\,10^{-3}$ | 81.5% |

Table 4: Summary of the performances of the chosen models on the test set (durum wheat regression problem)

As in the previous section, we can rank the methods in increasing order of modelling error, we obtain the following result:

$$\text{G-S2} < \text{G-S1} < \text{G-O} < \text{L-O} \leq \text{L-S2} < \text{L-S1},$$

where G stands for Gaussian kernel and L for linear kernel (hence G-S2 stands for kernel ridge regression with gaussian kernel and smoothing splines with order 2 derivatives); $<$ corresponds to a significant difference (for a paired Student test with level 1%) and $\leq$ to a non significant one. For this application, there is a significant gain in using a non linear model (the Gaussian kernel). In addition, the use of derivatives leads to less contrasted performances that the ones obtained in the previous section but it still improves the quality of the non linear model in a significant way. In term of normalized mean squared error (mean squared error divided by the variance of the

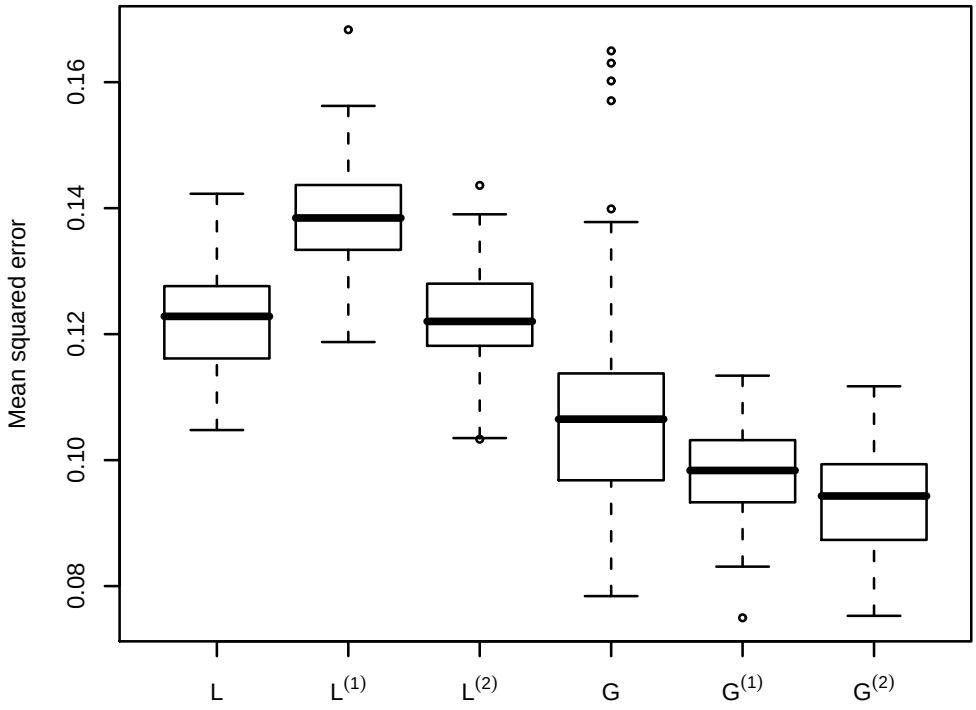


Figure 5: Mean squared error boxplots for the "durum wheat" regression problem (see Table 4 for the full names of the regression models)

target variable), using a non linear model with the second derivatives of the spectra corresponds to an average gain of more than 5% (i.e., a reduction of the normalised mean squared error from 24% for the standard linear model to 18.6%).

## 7. Conclusion

In this paper we proposed a theoretical analysis of a common practice that consists in using derivatives in classification or regression problems when the predictors are curves. Our method relies on smoothing splines reconstruction of the functions which are known only via a discrete deterministic sampling. The method is proved to be consistent for very general classifiers or regression schemes: it reaches asymptotically the best risk that could have been obtained by constructing a regression/classification model on the true random functions.

We have validated the approach by combining it with nonparametric regression and classification algorithms to study two real-world spectrometric datasets. The results obtained in these applications confirm once again that relying on derivatives can improve the quality of predictive models compared to a direct use of the sampled functions. The way the derivatives are estimated does not have a strong impact on the performances except when the data are noisy. In this case, the use of smoothing splines is quite relevant.

In the future, several issues could be addressed. An important practical problem is the choice of the best order of the derivative, $m$. We consider that a model selection approach relying on a penalized error loss could be used, as is done, in e.g., Rossi and Villa (2006), to select the dimension of truncated basis representation for functional data. Note that in practice, such parameter selection method could lead to select $m = 0$ and therefore to automatically exclude derivative calculation when it is not needed. This will extend the application range of the proposed model.

A second important point to study it the convergence rate for the method. It would be very convenient for instance, to be able to relate the size of the sampling grid to the number of functions. But, this latter issue would require the use of additional assumptions on the smoothness of the regression function whereas the result presented in this paper, even if more limited, only needs mild conditions.

## 8. Acknowledgement

We thank Cécile Levasseur and Sylvain Coulomb (École d'Ingénieurs de Purpan, EIP, Toulouse, France) for sharing the interesting problem presented in Section 6.3.

We also thank Philippe Besse (Institut de Mathématiques de Toulouse, Université de Toulouse, France) for helpfull discussions and suggestions.

Finally, we thank the anonymous reviewers for their valuable comments and suggestions that helped to improve the quality of the paper.

## References

Bahlmann, C., Burkhardt, H., 2004. The writer independent online handwriting recognition system *frog on hand* and cluster generative statistical dynamic time warping. IEEE Transactions on Pattern Analysis and Machine Intelligence 26, 299–310.

Berlinet, A., Biau, G., Rouvière, L., 2008. Functional supervised classification with wavelets. Annales de l'ISUP 52, 61–80.

Berlinet, A., Thomas-Agnan, C., 2004. Reproducing Kernel Hilbert Spaces in Probability and Statistics. Kluwer Academic Publisher.

Besse, P., Ramsay, J., 1986. Principal component analysis of sampled curves. Psychometrika 51, 285–311.

Biau, G., Bunea, F., Wegkamp, M., 2005. Functional classification in Hilbert spaces. IEEE Transactions on Information Theory 51, 2163–2172.

Cardot, H., Ferraty, F., Sarda, P., 1999. Functional linear model. Statistics and Probability Letters 45, 11–22.

Cawley, G., Talbot, N., 2004. Fast exact leave-one-out cross-validation of sparse least-squares support vector machines. Neural Networks 17, 1467–1475.

Cox, D., 1984. Multivariate smoothing splines functions. SIAM Journal on Numerical Analysis 21, 789–813.

Craven, P., Wahba, G., 1978. Smoothing noisy data with spline functions. Numerische Mathematik 31, 377–403.

Dauxois, J., Pousse, A., 1976. Les analyses factorielles en calcul des probabilités et en statistique : essai d'étude synthétique. Thèse d'État. Université Toulouse III.

Dejean, S., Martin, P., Baccini, A., Besse, P., 2007. Clustering time-series gene expression data using smoothing spline derivatives. EURASIP Journal on Bioinformatics and Systems Biology 2007, Article ID70561.

Deville, J., 1974. Méthodes statistiques et numériques de l'analyse harmonique. Annales de l'INSEE 15, 3–97.

Devroye, L., Györfi, L., 1985. Nonparametric Density Estimation: the $L_1$ view. John Wiley, New York.

Devroye, L., Györfi, L., Krzyżak, A., Lugosi, G., 1994. On the strong universal consistancy of nearest neighbor regression function estimates. The Annals of Statistics 22, 1371–1385.

Devroye, L., Györfi, L., Lugosi, G., 1996. A Probabilistic Theory for Pattern Recognition. Springer-Verlag, New York.

Devroye, L., Krzyżak, A., 1989. An equivalence theorem for $l_1$ convergence of the kernel regression estimate. Journal of Statistical Planning and Inference 23, 71–82.

Faragó, T., Györfi, L., 1975. On the continuity of the error distortion function for multiple-hypothesis decisions. IEEE Transactions on Information Theory 21, 458–460.

Ferraty, F., Vieu, P., 2003. Curves discrimination: a non parametric approach. Computational and Statistical Data Analysis 44, 161–173.

Ferraty, F., Vieu, P., 2006. NonParametric Functional Data Analysis. Springer.

Ferré, L., Yao, A., 2003. Functional sliced inverse regression analysis. Statistics 37, 475–488.

Györfi, L., Kohler, M., Krzyżak, A., Walk, H., 2002. A Distribution-Free Theory of Nonparametric Regression. Springer, New York.

Heckman, N., Ramsay, J., 2000. Penalized regression with model-based penalties. The Canadian Journal of Statistics 28, 241–258.

James, G., 2002. Generalized linear models with functional predictor variables. Journal of the Royal Statistical Society Series B 64, 411–432.

James, G., Hastie, T., 2001. Functional linear discriminant analysis for irregularly sampled curves. Journal of the Royal Statistical Society, Series B 63, 533–550.

James, G., Hastie, T., Sugar, C., 2000. Principal component models for sparse functional data. Biometrika 87, 587–602.

James, G., Silverman, B., 2005. Functional adaptive model estimation. Journal of the American Statistical Association 100, 565–576.

Kallenberg, O., 1997. Foundations of Modern Probability. Probability and its Applications, Spinger.

Kimeldorf, G., Wahba, G., 1971. Some results on Tchebycheffian spline functions. Journal of Mathematical Analysis and Applications 33, 82–95.

Laloë, T., 2008. A k-nearest neighbor approach for functional regression. Statistics and Probability Letters 78, 1189–1193.

Lugosi, G., Zeger, K., 1990. Nonparametric estimation via empirical risk minimization. IEEE Transaction on Information Theory 41, 677–687.

Mas, A., Pumo, B., 2009. Functional linear regression with derivatives. Journal of Nonparametric Statistics 21, 19–40. Submitted: under revision. Available at `http://www.math.univ-montp2.fr/~mas/FLRD.pdf`.

Pollard, D., 2002. A User's Guide to Measure Theoretic Probability. Cambridge University Press, Cambridge.

Ragozin, D., 1983. Error bounds for derivative estimation based on spline smoothing of exact or noisy data. Journal of Approximation Theory 37, 335–355.

Ramsay, J., Dalzell, C., 1991. Some tools for functional data analysis (with discussion). Journal of the Royal Statistical Society. Series B. Statistical Methodology 53, 539–572.

Ramsay, J., Silverman, B., 1997. Functional Data Analysis. Springer Verlag, New York.

Ramsay, J., Silverman, B., 2002. Applied Functional Data Analysis. Springer Verlag.

Rossi, F., Conan-Guez, B., 2005. Functional multi-layer perceptron: a non-linear tool for functional data anlysis. Neural Networks 18, 45–60.

Rossi, F., Conan-Guez, B., 2006. Theoretical properties of projection based multilayer perceptrons with functional inputs. Neural Processing Letters 23, 55–70.

Rossi, F., Delannay, N., Conan-Guez, B., Verleysen, M., 2005. Representation of functional data in neural networks. Neurocomputing 64, 183–210.

Rossi, F., Villa, N., 2006. Support vector machine for functional data classification. Neurocomputing 69, 730–742.

Saunders, G., Gammerman, A., Vovk, V., 1998. Ridge regression learning algorithm in dual variables, in: Proceedings of the Fifteenth International Conference on Machine Learning (ICML'98), Madison, Wisconsin, USA. pp. 515–521.

Shawe-Taylor, J., Cristianini, N., 2004. Kernel methods for pattern analysis. Cambridge University Press, Cambridge, UK.

Steinwart, I., 2002. Support vector machines are universally consistent. Journal of Complexity 18, 768–791.

Steinwart, I., Christmann, A., 2008. Support Vector Machines. Information Science and Statistics, Springer.

Thodberg, H., 1996. A review of bayesian neural network with an application to near infrared spectroscopy. IEEE Transaction on Neural Networks 7, 56–72.

Utreras, F., 1981. Optimal smoothing of noisy data using spline functions. SIAM Journal on Scientific Computing 2, 153–163.

Utreras, F., 1988. Boundary effects on convergence rates for Tikhonov regularization. Journal of Approximation Theory 54, 235–249.

Wahba, G., 1990. Spline Models for Observational Data. Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania.

Williams, B., Toussaint, M., Storkey, A., 2006. Extracting motion primitives from natural handwriting data, in: In Proceedings of the International Conference on Artificial Neural Networks (ICANN).

Zhao, L., 1987. Exponential bounds of mean error for the nearest neighbor estimates of regression functions. Journal of Multivariate Analysis 21, 168–178.

## 9. Proofs

### *9.1. Theorem 1*

In the original theorem (Lemma 3.1) in Kimeldorf and Wahba (1971), one has to verify that $(k_0(t_l,.))_l$ spans $\mathcal{H}_0^m$ and that $(k_1(t_l,.))_l$ are linearly independent. These are consequences of Assumption (A1).

First, $k_0(s,t) = \sum_{i,j=0}^{m-1} b_{ij}^{(-1)} s^i t^j$ where $\widetilde{B} = (b_{i,j}^{(-1)})_{i,j}$ is the inverse of $(\sum_{l=1}^m B^l s^i B^l t^j)_{i,j}$ (see Heckman and Ramsay (2000)). Then $(k_0(t_1,s),\ldots,k_0(t_{|\tau_d|},s)) = (1,s,\ldots,s^{m-1})\widetilde{B}[V_{m-1}(t_1,\ldots,t_{|\tau_d|})]^T$ where $V_{m-1}(t_1,\ldots,t_{|\tau_d|})$ is the Vandermonde matrix with $m-1$ columns and $|\tau_d|$ rows associated to values $t_1,\ldots,t_{|\tau_d|}$. If the $(t_l)_l$ are distinct, this matrix is of full rank.

Moreover the reproducing property shows that $\sum_{l=1}^{|\tau_d|} a_l k_1(t_l,.) \equiv 0$ implies $\sum_{l=1}^{|\tau_d|} a_l f(t_l) \equiv 0$ for all $f \in \mathcal{H}_1^m$. Hence, $\mathcal{H}_1^m = \mathrm{Ker}\left(B^T, \sum_{l=1}^{\tau_d} a_l\zeta_l\right)^T$ where $\zeta_l$ denotes the linear form $h \in \mathcal{H}^m \to h(t_l)$. As the co-dimension of $\mathcal{H}_1^m$ is $\dim\mathcal{H}_0^m = m$ and as, by Assumption (A1), $B$ is linearly independent of $\sum_{l=1}^{\tau_d} a_l\zeta_l$, we thus have $\sum_{l=1}^{\tau_d} a_l\zeta_l \equiv 0$ (or $\mathrm{codim}\,\mathrm{Ker}\left(B^T, \sum_{l=1}^{\tau_d} a_l\zeta_l\right)^T = \dim \mathrm{Im}\left(B^T, \sum_{l=1}^{\tau_d} a_l\zeta_l\right)$ would be $m+1$). Thus, we obtain that $\sum_{l=1}^{|\tau_d|} a_l f(t_l) \equiv 0$ for all $f$ in $\mathcal{H}^m$ and, as $(t_l)$ are distinct, that $a_l = 0$ for all $l$, leading to the independence conclusion for the $(k_1(t_l,.))_l$.

Finally, we prove that $\mathcal{S}_{\lambda,\tau_d}$ is of full rank. Indeed, if $\mathcal{S}_{\lambda,\tau_d}\mathbf{x}^{\tau_d} = 0$, $\omega^T M_0 \mathbf{x}^{\tau_d} = 0$ and $\eta^T M_1 \mathbf{x}^{\tau_d} = 0$. As $(\omega_k)_k$ is a basis of $\mathcal{H}_0^m$, $\omega^T M_0 \mathbf{x}^{\tau_d} = 0$ implies $M_0\mathbf{x}^{\tau_d} = 0$ and therefore $M_1 = (K_1 + \lambda I_d)^{-1}$. As shown above, the $(k_1(t_l,.))_l$ are linearly independent and therefore $\eta M_1 \mathbf{x}^{\tau_d} = 0$ implies $M_1\mathbf{x}^{\tau_d} = 0$, which in turns leads to $\mathbf{x}^{\tau_d} = 0$ via the simplified formula for $M_1$.

*9.2. Corollary 1*

We give only the proof for the classification case, the regression case is identical.

According to Theorem 1, there is a full rank linear mapping from $\mathbb{R}^{|\tau_d|}$ to $\mathcal{H}^m$, $\mathcal{S}_{\lambda,\tau_d}$, such that for any function $x \in \mathcal{H}^m$, $\widehat{x}_{\lambda,\tau_d} = \mathcal{S}_{\lambda,\tau_d}\mathbf{x}^{\tau_d}$. Let us denote $\mathcal{I}_{\lambda,\tau_d}$ the image of $\mathbb{R}^{|\tau_d|}$ by $\mathcal{S}_{\lambda,\tau_d}$, $\mathbf{P}_{\lambda,\tau_d}$ the orthogonal projection from $\mathcal{H}^m$ to $\mathcal{I}_{\lambda,\tau_d}$ and $\mathcal{S}^{-1}_{\lambda,\tau_d}$ the inverse of $\mathcal{S}_{\lambda,\tau_d}$ on $\mathcal{I}_{\lambda,\tau_d}$. Obviously, we have $\mathcal{S}^{-1}_{\lambda,\tau_d} \circ \mathbf{P}_{\lambda,\tau_d}(\widehat{x}_{\lambda,\tau_d}) = \mathbf{x}^{\tau_d}$.

Let $\psi$ be a measurable function from $\mathbb{R}^{|\tau_d|}$ to $\{-1,1\}$. Then $\zeta_\psi$ defined on $\mathcal{H}^m$ by $\zeta_\psi(u) = \psi\left(\mathcal{S}^{-1}_{\lambda,\tau_d} \circ \mathbf{P}_{\lambda,\tau_d}(u)\right)$ is a measurable function from $\mathcal{H}^m$ to $\{-1,1\}$ (because $\mathcal{S}^{-1}_{\lambda,\tau_d}$ and $\mathbf{P}_{\lambda,\tau_d}$ are both continuous). Then for any measurable $\psi$, $\inf_{\phi:\mathcal{H}^m\to\{-1,1\}} \mathbb{P}\left(\phi(\widehat{X}_{\lambda,\tau_d}) \neq Y\right) \leq \mathbb{P}\left(\zeta_\psi(\widehat{X}_{\lambda,\tau_d}) \neq Y\right) = \mathbb{P}\left(\psi(\mathbf{X}^{\tau_d}) \neq Y\right)$, and therefore

$$\inf_{\phi:\mathcal{H}^m\to\{-1,1\}} \mathbb{P}\left(\phi(\widehat{X}_{\lambda,\tau_d}) \neq Y\right) \leq \inf_{\phi:\mathbb{R}^{|\tau_d|}\to\{-1,1\}} \mathbb{P}\left(\phi(\mathbf{X}^{\tau_d}) \neq Y\right). \tag{10}$$

Conversely, let $\psi$ be a measurable function from $\mathcal{H}^m$ to $\{-1,1\}$. Then $\zeta_\psi$ defined on $\mathbb{R}^{|\tau_d|}$ by $\zeta_\psi(\mathbf{u}) = \psi(\mathcal{S}_{\lambda,\tau_d}(\mathbf{u}))$, is measurable. Then for any measurable $\psi$, $\inf_{\phi:\mathbb{R}^{|\tau_d|}\to\{-1,1\}} \mathbb{P}\left(\phi(\mathbf{X}^{\tau_d}) \neq Y\right) \leq \mathbb{P}\left(\zeta_\psi(\mathbf{X}^{\tau_d}) \neq Y\right) = \mathbb{P}\left(\psi(\widehat{X}_{\lambda,\tau_d}) \neq Y\right)$, and therefore

$$inf_{\phi:\mathbb{R}^{|\tau_d|}\to\{-1,1\}} \mathbb{P}\left(\phi(\mathbf{X}^{\tau_d}) \neq Y\right) \leq \inf_{\phi:\mathcal{H}^m\to\{-1,1\}} \mathbb{P}\left(\phi(\widehat{X}_{\lambda,\tau_d}) \neq Y\right). \tag{11}$$

The combination of equations (10) and (11) gives equality (4).

*9.3. Corollary 3*

1. **Suppose assumption (A4a) is fullfilled**
   The proof is based on Theorem 1 in Faragó and Györfi (1975). This theorem relates the Bayes risk of a classification problem based on $(X, Y)$ with the Bayes risk of the problem $(T_d(X), Y)$ where $(T_d)$ is a series of transformations on $X$.

   More formally, for a pair of random variables $(X, Y)$, where $X$ takes values in $\mathcal{X}$, an arbitrary metric space, and $Y$ in $\{-1, 1\}$, let us

denote for any series of functions $T_d$ from $\mathcal{X}$ to itself, $L^*(T_d) = \inf_{\phi:\mathcal{X}\to\{-1,1\}} \mathbb{P}(\phi(T_d(X)) \neq Y)$. Theorem 1 from Faragó and Györfi (1975) states that $\mathbb{E}(\delta(T_d(X), X)) \xrightarrow{d\to+\infty} 0$ implies $L^*(T_d) \xrightarrow{d\to+\infty} L^*$, where $\delta$ denotes the metric on $\mathcal{X}$.

This can be applied to $\mathcal{X} = (\mathcal{H}^m, \langle ., . \rangle_{L^2})$ with $T_d(X) = \widehat{X}_{\lambda_d,\tau_d} = S_{\lambda_d,\tau_d}\mathbf{X}^{\tau_d}$: under Assumptions (A1) and (A2), Theorem 2 gives: $\|T_d(X) - X\|_{L^2}^2 \leq \left(A_{R,m}\lambda_d + B_{R,m}\frac{1}{|\tau_d|^{2m}}\right)\|D^m X\|_{L^2}^2$. Taking the expectation of both sides gives $\mathbb{E}(\|T_d(X) - X\|_{L^2}) \leq \left(A_{R,m}\lambda_d + B_{R,m}\frac{1}{|\tau_d|^{2m}}\right)\mathbb{E}\left(\|D^m X\|_{L^2}^2\right)$, using the fact that the constants are independent of the function under analysis. Then under Assumptions (A4a) and (A3), $\mathbb{E}(\|T_d(X) - X\|_{L^2}) \xrightarrow{d\to+\infty} 0$. According to Faragó and Györfi (1975), this implies $\lim_{d\to\infty} L_d^* = L^*$.

2. **Suppose assumption (A4b) is fullfilled**
The conclusion will follow both for classification case and for regression case. The proof follows the general ideas of Biau et al. (2005); Rossi and Conan-Guez (2006); Rossi and Villa (2006); Laloë (2008). Under assumption (A1), by Theorem 1 and with an argument similar to those developed in the proof of Corollary 1, $\sigma(\widehat{X}_{\lambda_d,\tau_d}) = \sigma(\{X(t)\}_{t\in\tau_d})$. From assumption (A4b), $\sigma(\{X(t)\}_{t\in\tau_d})$ is clearly a filtration. Moreover, as $\mathbb{E}(Y)$ and thus $\mathbb{E}(Y^2)$ are finite, $\mathbb{E}\left(Y|\widehat{X}_{\lambda_d,\tau_d}\right)$ is a uniformly bounded martingal for this filtration (see Lemma 35 of Pollard (2002)). This martingale converges in $L^1$-norm to $\mathbb{E}\left(Y|\sigma\left(\cup_d\sigma(\widehat{X}_{\lambda_d,\tau_d})\right)\right)$; we have

- $\sigma\left(\cup_d\sigma(\widehat{X}_{\lambda_d,\tau_d})\right) \subset \sigma(X)$ as $\widehat{X}_{\lambda_d,\tau_d}$ is a function of $X$ (via Theorem 1);
- by Theorem 2, $\widehat{X}_{\lambda_d,\tau_d} \xrightarrow{d\to+\infty,\text{ surely}} X$ in $L^2$ which proves that $X$ is $\sigma\left(\cup_d\sigma(\widehat{X}_{\lambda_d,\tau_d})\right)$-measurable.

Finally, $\mathbb{E}\left(Y|\sigma\left(\cup_d\sigma(\widehat{X}_{\lambda_d,\tau_d})\right)\right) = \mathbb{E}(Y|X)$ and $\mathbb{E}\left(Y|\widehat{X}_{\lambda_d,\tau_d}\right) \xrightarrow{d\to+\infty,\ L^1} \mathbb{E}(Y|X)$.

The conclusion follows from the fact that:

(a) *binary classification case:* the bound $L_d^* - L^* \leq 2\mathbb{E}\left(\left|\mathbb{E}\left(Y|\widehat{X}_{\lambda_d,\tau_d}\right) - \mathbb{E}\left(Y|X\right)\right|\right)$ (see Theorem 2.2 of Devroye et al. (1996)) concludes the proof;

(b) *regression case:* as $\mathbb{E}\left(Y^2\right)$ is finite, $\mathbb{E}\left(\mathbb{E}\left(Y|\widehat{X}_{\lambda_d,\tau_d}\right)^2\right)$ is also finite and the convergence also happens for the quadratic norm (see Corollary 6.22 in Kallenberg (1997)), i.e.,

$$\lim_{d\to+\infty} \mathbb{E}\left(\left(\mathbb{E}\left(Y|X\right) - \mathbb{E}\left(Y|\widehat{X}_{\lambda_d,\tau_d}\right)\right)^2\right) = 0$$

Hence, as $L_d^* - L^* = \mathbb{E}\left(\left(\mathbb{E}\left(Y|X\right) - \mathbb{E}\left(Y|\widehat{X}_{\lambda_d,\tau_d}\right)\right)^2\right)$, the conclusion follows.

*9.4. Theorem 3*

We have

$$L(\phi_{n,d}) - L^* = L\phi_{n,\tau_d} - L_d^* + L_d^* - L^*. \tag{12}$$

Let $\epsilon$ be a positive real. By Corollary 3, it exists $d_0 \in \mathbb{N}^*$ such that, for all $d \geq d_0$,

$$L_d^* - L^* \leq \epsilon. \tag{13}$$

Moreover, as shown in Corollary 1 and as $\mathbf{Q}_{\lambda_d,\tau_d}$ is invertible, we have in the binary classification case: $L_d^* = \inf_{\phi:\mathbb{R}^{|\tau_d|}\to\{-1,1\}} \mathbb{P}\left(\phi(\mathbf{X}^{\tau_d}) \neq Y\right) = \inf_{\phi:\mathbb{R}^{|\tau_d|}\to\{-1,1\}} \mathbb{P}\left(\phi\left(\mathbf{Q}_{\lambda_d,\tau_d}\mathbf{X}^{\tau_d}\right) \neq Y\right)$, and in the regression case: $L_d^* = \inf_{\phi:\mathbb{R}^{|\tau_d|}\to\mathbb{R}} \mathbb{E}\left(\left[\phi\left(\mathbf{X}^{\tau_d}\right) - Y\right]^2\right) = \inf_{\phi:\mathbb{R}^{|\tau_d|}\to\mathbb{R}} \mathbb{E}\left(\left[\phi\left(\mathbf{Q}_{\lambda_d,\tau_d}\mathbf{X}^{\tau_d}\right) - Y\right]^2\right)$. By hypothesis, for any fixed $d$, $\phi_{n,\tau_d}$ is consistent, that is

$$\lim_{n\to+\infty} \mathbb{E}\left(L(\phi_{n,\tau_d})\right) = \inf_{\phi:\mathbb{R}^{|\tau_d|}\to\{-1,1\}} \mathbb{P}\left(\phi\left(\mathbf{Q}_{\lambda_d,\tau_d}\mathbf{X}^{\tau_d}\right) \neq Y\right),$$

in the classification case and

$$\lim_{n\to+\infty} \mathbb{E}\left(L(\phi_{n,\tau_d})\right) = \inf_{\phi:\mathbb{R}^{|\tau_d|}\to\mathbb{R}} \mathbb{E}\left(\left[\phi\left(\mathbf{Q}_{\lambda_d,\tau_d}\mathbf{X}^{\tau_d}\right) - Y\right]^2\right),$$

in the regression case, and therefore for any fixed $d_0$, $\lim_{n\to+\infty} \mathbb{E}\left(L(\phi_{n,\tau_{d_0}})\right) = L_{d_0}^*$. Combined with equations (12) and (13), this concludes the proof.