Functional statistical classification of non-linear dynamics and random surfaces roughness in control systems

Article *in* International Journal of Mathematical Models and Methods in Applied Sciences · January 2015

CITATION
1

READS
88

2 authors:

Javier Álvarez Liébana
Complutense University of Madrid
22 PUBLICATIONS **81** CITATIONS
SEE PROFILE

María Dolores Ruiz-Medina
University of Granada
163 PUBLICATIONS **1,706** CITATIONS
SEE PROFILE

Functional statistical classification of non-linear dynamics and random surfaces roughness in control systems

Javier Álvarez-Liébana and M. Dolores Ruiz-Medina

***Abstract*—** This paper addresses, in a nonparametric functional statistical framework, the problem of classification of nonlinear features of curve and surface data in control systems. Specifically, on the one hand, in the detection of nonlinear dynamic features, wavelength absorbance curve data are analyzed for different meat pieces to discriminate between two categories of meat in quality control in food industry. On the other hand, in the nonparametric functional classification of deterministic and random surface roughness and irregularities, in the field of railway engineering, train deterministic and random vibrations are analyzed to discriminate between different nonlinear features characterizing roughness and irregularities of railway.

***Keywords*—** Functional nonparametric classification, Irregularity of railway track, n-dimensional numerical integration, Random surfaces, Random vibrations, Smolyak quadrature.

I. INTRODUCTION

Non-linear dynamics and features in the data can be captured and suitable analyzed within the Functional Statistical framework. Temporal and spatial functional statistics are relatively recent branches of Statistics, where non-parametric statistical techniques are now been developing to approximate the nonlinear functional form of the probability distribution underlying to a sequence of random curves, surfaces, etc. In this context, new criteria for curve classification are proposed (see [8, 31], among others).

These procedures for random curve classification are designed in the absence or in the presence of interactions between different individuals, as well as between different times (see [1, 8, 16, 18, 24, 26]). In [18], a variant of linear discriminant analysis, in terms of the curve projections assuming a Gaussian distribution with common covariance matrix for all classes, is considered in the setting of filtering methods. Specifically, minimization of the distance to the group mean is the criterion adopted in this functional classification methodology. In [16] a likelihood-based approach based on quadratic discriminant analysis is presented. They propose a fully nonparametric density estimation, and, in practice, multivariate Gaussian densities are considered. Dealing with nonlinear discriminant algorithms, the learning optimal kernel for kernel Fisher discriminant analysis (KFDA) is proposed in [10] to be able to optimize a combination of weight coefficients and kernels. In a generalized linear model framework, the model-based functional classification procedures proposed in [23, 32] are implemented. Specifically, for dimension reduction, Functional Principal Component Analysis (FPCA), and local wavelet-vaguelette decomposition are considered. K-nearest neighbor method is applied to Fourier coefficients in [5]. Wavelet bases are selected for projection in [4]. In [19] spline bases are considered in a random effect model context, combining the best properties of filtering and regularization methods. These methods are effective when the observations are sparse, irregularly spaced or occur at different time points for each subject (see also [2], where B-splines bases are previously chosen for projection in the application of k-means-based classification procedure). In [14], a support vector machine is used to scene classification, in order to construct an effective clustering procedure for real time applications, in particular, for image sequence classification depending on several factors.

Functional nonparametric statistical classification procedures, based on kernels, are extensively developed in the context of statistical learning methods (see, for example, [39]). In this framework, the unknown function is estimated, considering its optimal approximation in a functional class given by a Reproducing Kernel Hilbert Space (RKHS), under some prescribed criterion. Chaos game representation and multifractal analysis can also be considered in the classification of functional protein sequences displaying singular features (see, for instance, [45, 46]).

This paper deals with the functional statistical nonparametric classification of non-linear random functions with n-dimensional support (e.g., curves, surfaces, etc.). They are assumed to be uncorrelated random functions. As motivation

This work has been supported in part by project MTM2012-32674 (co-funded with FEDER funds).

Javier Álvarez-Liébana is PhD student in the University of Granada, Department of Statistics and O.R., Campus Fuente Nueva s/n, 18071 Granada, Spain (email: javialvaliebana@ugr.es).

M. Dolores Ruiz-Medina is Full Professor in the University of Granada, Department of Statistics and O.R., Campus Fuente Nueva s/n, 18071 Granada, Spain (email: mruiz@ugr.es).

for illustration of the proposed functional nonparametric statistical methodology we address two problems in the applied areas of food industry and railway engineering. Specifically, fat content is first analyzed for classification of meat pieces, from the observation of spectrometric curve data corresponding to the absorbance measured at 100 wavelengths. On the other hand, in the random surface discrimination context, the statistical analysis of train deterministic and random vibrations is achieved from the nonparametric functional statistical classification of rail roughness and irregularities. The results obtained, after the implementation of the proposed classification methodology are showed in Sections IV. and VI., respectively. In such an implementation, an extended version of the classification algorithm formulated in [8] is derived. Namely, numerical integration is performed by applying Smolyak quadrature rule, after interpolation over a finer n - dimensional grid the values observed at a coarser grid, which constitutes our actual functional dataset. Different semi-metrics can then be applied, mainly based on FPCA and Functional Partial Least Squares Regression (FPLSR), which is an extension of Partial Least Squares technique (see, e.g., [29]). In addition, the kernel estimation of the posterior probability of belonging to each one of the categories defining the response provides us a rule for classification of the observed n - dimensional supported functional data in a nonparametric statistical context.

The resulting classification procedure for non-linear random functions with n-dimensional support, in the context of nonparametric functional statistics, allows discrimination in a more flexible framework. In particular, this paper provides an extension to the two-dimensional case of the one-dimensional models proposed in [9, 25, 47] for the analysis of imperfections of railway track. These irregularities are the second source of bridge vibrations and the first one of train vibrations, and can be classified into non-random and random irregularities (as the roughness of the rails). The dynamics of these railway tracks under moving trains must be taken into account in order to construct and design the railway bridges and beams, as well as to locate and construct the railway stations and surrounding buildings. The effects of rail roughness and rail irregularities on the dynamic behavior of bridge and vehicles are considered in [25, 47]. In this paper, the non-random imperfections are represented in terms of a two-dimensional function perturbed by Gaussian white noise, reflecting the measurement device error, while the random ones will be defined in terms of zero-mean Gaussian random surfaces, displaying different non-linear spatial patterns according to their spatial correlation structure.

The outline of the paper is as follows. Section II. presents some preliminaries definitions and elements involved in the functional statistical nonparametric classification algorithm studied in [8]. Section III. establishes the main steps of the proposed classification algorithm for n-dimensional supported non-linear random functions, and in particular, for random curves and surfaces. The application of this algorithm to spectrometric curve data for meat piece classification according to fat content is illustrated in Section IV.. Section V. provides a training simulation study to discriminate between different trend surfaces in Gaussian random surface classification. A simulation study is undertaken in Section VI. for illustration of the proposed functional classification methodology for perturbed deterministic and random irregularities in the surface of railway track. Conclusions are drawn in Section VII..

II. PRELIMINARIES ABOUT FUNCTIONAL NONPARAMETRIC CLASSIFICATION

Let us first introduce the preliminary elements and definitions, as well as the needed notation required for the description of the curve statistical functional classification algorithm proposed by [8] in a nonparametric framework.

Assume that $T = (t_{min}, t_{max})$ is an interval in $\mathbb{R}$. We shall use the notation:

- $\boldsymbol{\chi} = \{\boldsymbol{\chi}(t); t \in T\}$ for representing a functional random variable (f.r.v.), that is, a random variable $\boldsymbol{\chi}$ that takes values in an infinite dimensional space.
- χ functional data (f.d.) denotes an observation of $\boldsymbol{\chi}$.
- We shall denote a funcional dataset (f.dat.) $\{\chi_i\}_{i=1,\ldots,n}$ as the observation of n-sample f.r.v $\{\boldsymbol{\chi}_i\}_{i=1,\ldots,n} \sim \boldsymbol{\chi}$.

Different families of semi-metrics mainly based on FPCA (see [17] among others), FPLSR, and derivatives are commonly used to measure distances between curves. In the context of infinite dimensional spaces, they are usually computed by numerical integration, considering, in our case, n-dimensional integration based on suitable quadrature rules.

II..I. FUNCTIONAL PRINCIPAL COMPONENT ANALYSIS (FPCA)

This technique is based on projection into the eigenvector system of the covariance operator, obtaining a series expansion of the f.r.v. defining our data set, in terms of uncorrelated r.v., with scale parameters given by the square root of the associated eigenvalues. It is well-known that PCA (with euclidean metric) is formulated as follows:

$$z_i = \frac{\langle \boldsymbol{v}_i, \boldsymbol{x} \rangle}{\|\boldsymbol{v}_i\|} = \frac{1}{\|\boldsymbol{v}_i\|} \sum_{j=1}^{p} v_{ij} x_j = \frac{1}{\|\boldsymbol{v}_i\|} \boldsymbol{v}_i^T \boldsymbol{x} \quad (1)$$

$$\boldsymbol{x} = \sum_{j=1}^{p} \boldsymbol{e}_j x_j = \sum_{j=1}^{p} \boldsymbol{v}_j z_j, \quad (2)$$

where, for $i = 1, \ldots, p$, $E\left[\boldsymbol{z}_i^2\right] = \lambda_i$, with $\lambda_1 \geq \ldots \geq \lambda_p$.

In the infinite-dimensional case, we consider the spaces L^p with respect to a measure μ, introduced in terms of the seminorm $\|\cdot\|_p$, given by

$$\|f\|_p := \left(\int |f(x)|^p \mu(dx) \right)^{\frac{1}{p}}. \quad (3)$$

In particular, we concentrate in the case of $p = 2$, where we have a Hilbert space structure. Recall the fundamental definitions associated with this case.

Definition 1 (Eigenfunctions) *Let A be a linear operator, a function $f \neq 0$ is an eigenfunction of A if $Af = \lambda f$.*

Definition 2 (Orthogonal functions) *Let $(H, \langle .,. \rangle)$ be a real valued pre-Hilbert space with the inner product*

$$\langle f, g \rangle_{\boldsymbol{w}} = \int f(x) g(x) \boldsymbol{w}(x) dx, \quad \forall f, g \in H,$$

where $\boldsymbol{w}$ is a weight function. Two functions f, g are then orthogonal iff $\langle f, g \rangle_{\boldsymbol{w}} = 0$.

The resulting series expansions in PCA (on left) and FPCA (on right) are given as follows, when $\{\boldsymbol{v}_j\}$ are normalized:

$$z_j = \langle \boldsymbol{v}_j, \boldsymbol{x} \rangle \qquad z_j = \int \boldsymbol{\chi}(x) v_j(x) dx \tag{4}$$

$$\boldsymbol{x} = \sum_{j=1}^{p} \boldsymbol{v}_j z_j \qquad \boldsymbol{\chi}(x) = \sum_{j=1}^{\infty} v_j(x) z_j \tag{5}$$

Thus, for the infinite-dimensional case we have

$$\boldsymbol{\chi}(z) =^{(5)} \sum_{j=1}^{\infty} \left(\int \boldsymbol{\chi}(x) v_j(x) dx \right) v_j(z), \tag{6}$$

and its truncated version can be written as

$$\widehat{\boldsymbol{\chi}}^{(q)}(z) =^{(6)} \sum_{j=1}^{q} \left(\int \boldsymbol{\chi}(x) v_j(x) dx \right) v_j(z). \tag{7}$$

From (7), the following semi-norm can be defined:

$$d_q^{FPCA}(\boldsymbol{\chi}_1, \boldsymbol{\chi}_2) = \sqrt{\sum_{j=1}^{q} \left(\int [\boldsymbol{\chi}_1 - \boldsymbol{\chi}_2](x) v_j(x) dx \right)^2} \tag{8}$$

From a practical point of view, the above integrals are approximated by a quadrature rule. Specifically, for the observed discretized curves, namely, $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, the following numerical approximation is computed:

$$d_q^{FPCA}(\boldsymbol{x}_1, \boldsymbol{x}_2) = \sqrt{\sum_{j=1}^{q} \left(\sum_{i=1}^{I} w_i [\boldsymbol{x}_1 - \boldsymbol{x}_2](t_i) v_{ji} \right)^2} \tag{9}$$

where t_i $(i = 1, \ldots, I)$ are the nodes, $1 \leq q \leq n$ the number of components chosen, $\boldsymbol{\Sigma}_{\boldsymbol{\chi}}(s,t) = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\chi}_i(s) \boldsymbol{\chi}_i(t)$ the empirical version of covariance kernel, i.e., its empirical matrix approximation, $\boldsymbol{v}_j = (v_{j1}, \ldots, v_{jI})$, $j = 1, \ldots, q$, are the empirical eigenvectors of $\mathbf{W}^{1/2} \boldsymbol{\Sigma} \mathbf{W}^{1/2}_{I \times I}$, with $\boldsymbol{W} = diag(w_1, \ldots, w_I)$ being a diagonal matrix with non-null entries given by the quadrature weights provided by a quadrature rule.

II..II. FUNCTIONAL PARTIAL LEAST SQUARES REGRESSION (FPLSR)

The Multivariate Partial Least Squares Regression (MPLSR) is an extension of PLSR motivated by dealing with multivariate response or when the number of predictors is very large in comparison with the number of observations.

We can apply MPLSR with only one scalar response but it would be inadequate with regard to the complexity of functional data. Hence, we are going to construct a multivariate response binary matrix where each column j represents if the i-th observation belongs to class j. Such as FPCA technique, we can extend MPLSR to FPLSR in functional framework, providing us g components depending on a number of factors q, which plays similar role to the number of dimensions retained in FPCA. The main difference between FPCA and FPLSR comes from the fact that the FPCA explains only the predictors, whereas the FPLSR approach computes a simultaneous decomposition of the set of predictors and responses, being able to explain both predictors and responses. Thus, we get a similar FPCA formula:

$$d_q^{FPLSR}(\boldsymbol{x}_1, \boldsymbol{x}_2) = \sqrt{\sum_{j=1}^{g} \left(\sum_{i=1}^{I} w_i [\boldsymbol{x}_1 - \boldsymbol{x}_2](t_i) v_{ji}^q \right)^2} \tag{10}$$

where $\boldsymbol{v}_1^q, \ldots, \boldsymbol{v}_g^q$ are performed by FPLSR.

II..III. SEMI-METRICS BASED ON DERIVATIVES

Lastly, we introduce the semi-metric based on derivatives. That is, the L^2 distance between the derivatives of different orders of two given curves is established as a measure of closeness in the following way:

$$d_q^{deriv}(\chi_1, \chi_2) = \sqrt{\int \left(\chi_1^{(q)} - \chi_2^{(q)} \right)^2 (x) dx} \tag{11}$$

where $\chi^{(q)}$ is the qth derivative of χ and $d_0^{deriv} = d_{L^2}$.

To avoid stability problems with derivatives, a B-spline basis approximation is usually considered (see, e.g., [6, 40]). Using the discretized curve $\boldsymbol{x}_i = (\chi_i(t_1), \ldots, \chi_i(t_I))$, we obtain the following approximation:

$$\widehat{\chi}_i(.) = \sum_{b=1}^{B} \widehat{\beta}_{ib} B_b(.) \quad \widehat{\chi}_i^{(q)}(.) = \sum_{b=1}^{B} \widehat{\beta}_{ib} B_b^{(q)}(.) \tag{12}$$

where $\{B_1, \ldots, B_B\}$ is a B-spline basis. Thus, for numerical approximation of

$$d_q^{deriv}(\boldsymbol{x}_1, \boldsymbol{x}_2) = \sqrt{\int \left(\widehat{\chi}_1^{(q)}(x) - \widehat{\chi}_2^{(q)}(x) \right)^2 dx}, \tag{13}$$

a quadrature rule is considered. Note that B-spline basis allow to work even with unbalanced data sets.

II..IV. NUMERICAL INTEGRATION: QUADRATURE RULES

To define all of these semi-metrics in a functional space, numerical integration in terms of a quadrature rules is required. Let see a brief about them.

There is a large variety of one-dimensional numerical integration procedures, as the trapezoidal rule [12], the Clenshaw-Curtis rule (see [12, 20, 28]) and Gauss rules introduced in [7, 11, 20]. We can also use stochastic simulation applying methods such as Monte Carlo (MC) and Quasi-Monte Carlo methods (QMC) (see, for example, [11]). We will restrict our attention to numerical integration, since a set of weights is needed.

According to [12, 20], in the following, we consider functions $f(x)$ from a regular function class:

$$\mathcal{C}^r(\Omega) := \left\{ f : \Omega \subset \mathbb{R}^n \to \mathbb{R}, \left\| \frac{\partial^s f}{\partial x^s} \right\|_\infty < \infty,\ s \leq r \right\}. \tag{14}$$

As we will see, the goal is to approximate the integral in a subset Ω of $\mathbb{R}^n$, $I^n f := \int_\Omega f(x)dx$ by a sequence of n_l -point quadrature ($n_l = 2^{l-1} + 1$).

II..V. FUNCTIONAL NONPARAMETRIC SUPERVISED CLASSIFICATION OF RANDOM CURVES

As described in [8], we now observe a f.r.v $\boldsymbol{\chi}$ and a categorical response $\boldsymbol{y}$ that represents the class membership of each element. The main aim is to be able to predict the class membership of a new f.d., by means of a nonparametric rule. Denoting by (E, d) a semi-metric space and $\overline{G} = \{1, \ldots, G\}$ a set of integers, we consider $(\boldsymbol{\chi}_i, \boldsymbol{y}_i)_{i=1,\ldots,n} \sim (\boldsymbol{\chi}, \boldsymbol{y})$ to be a sample of n independent pairs in $E \times \overline{G}$. Thus, (χ_i, y_i) denotes an observation of $(\boldsymbol{\chi}_i, \boldsymbol{y}_i)_{i=1,\ldots,n}$, and $(\boldsymbol{x}_i, y_i)$, with $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{iI})$ being the discretization of (χ_i, y_i).

Applying the Bayes rule, our goal is estimate $p_g(\chi) = P(Y = g|\boldsymbol{\chi} = \chi) = \mathbb{E}[\mathbb{1}_{Y=g}|\boldsymbol{\chi} = \chi]\ (g \in \overline{G})$, doing the assignment:

$$\widehat{y}(\chi) = arg \max_{g \in \overline{G}} \widehat{p}_g(\chi) \tag{15}$$

where $\widehat{\boldsymbol{p}}_{\boldsymbol{g}}(\chi) = (\widehat{p}_1(\chi), \ldots, \widehat{p}_G(\chi))$ are the estimated posterior probabilities and $\mathbb{1}_{Y=g}$ is the indicator function.

Let K be a kernel function and $\boldsymbol{\Lambda} : \mathbb{R}^{\mathbf{p}} \to \mathbb{R}$ a function (an operator in the infinite-dimensional case) which we want to estimate. We define the kernel smoother as:

$$K_h(\chi, \boldsymbol{\chi}_i) := K\left(\frac{d(\chi, \boldsymbol{\chi}_i)}{h(\chi)} \right), \tag{16}$$

where K is a positive kernel function that decreasing with the distance between $\boldsymbol{\chi}_i$ and χ, $h(\chi)$ is a positive bandwidth, depending on χ. Therefore, we can use the truncated kernel regression estimator of $\boldsymbol{\Lambda}$ proposed in [27, 43], in an infinite-dimensional setting, as follows:

$$\widehat{\boldsymbol{\Lambda}}(\chi) := \frac{\sum_{i=1}^n K_h(\chi, \boldsymbol{\chi}_i)\, \boldsymbol{\Lambda}(\boldsymbol{\chi}_i)}{\sum_{i=1}^n K_h(\chi, \boldsymbol{\chi}_i)}, \tag{17}$$

where $\boldsymbol{\Lambda}(\boldsymbol{\chi}_i) = \mathbb{E}[\mathbb{1}_{Y_i=g}|\chi_i = \boldsymbol{\chi}_i] = \mathbb{1}_{yi=g} = p_g(\boldsymbol{\chi}_i) = 1$. Thus, according to (17):

$$\widehat{p}_{g,h}(\chi) = \frac{\sum_{i=1}^n K\left(\frac{d(\chi, \boldsymbol{\chi}_i)}{h(\chi)} \right) \mathbb{1}_{yi=g}}{\sum_{i=1}^n K\left(\frac{d(\chi, \boldsymbol{\chi}_i)}{h(\chi)} \right)} = \sum_{\{i:y_i=g\}} w_{i,h}(\chi) \tag{18}$$

$$\text{with } w_{i,h}(\chi) = \frac{K\left(\frac{d(\chi, \boldsymbol{\chi}_i)}{h(\chi)} \right)}{\sum_{i=1}^n K\left(\frac{d(\chi, \boldsymbol{\chi}_i)}{h(\chi)} \right)}.$$

If we choose a kernel such that $K(x) = 0$ if $|x| < 1$ results:

$$\widehat{p}_{g,h}(\chi) = \sum_{i \in \mathcal{J}} w_{i,h}(\chi) \tag{19}$$

where $\mathcal{J} = \{i : y_i = g\} \cap \{i : d(\chi, \boldsymbol{\chi}_i) < h\}$.

II..VI. Bandwidth selection

Finally, we have to choose h with the goal of minimizing a loss function that depends on $\widehat{p}_{g,h}(\chi_i, y_i)$'s and y_i's:

$$h_{Loss} = arg \inf_h Loss(h). \tag{20}$$

With this aim, we will replace the choice of h among an infinite set $\mathcal{H}$ with an integer parameter k among a finite subset $\mathcal{K}$, by the consideration of k-Nearest Neighborhood (kNN) discretized version of (20):

$$\widehat{p}_{g,k}(\boldsymbol{x}) = \frac{\sum_{i \in \mathcal{J}} K\left(\frac{d(\boldsymbol{x}, \boldsymbol{x}_i)}{h_k(\boldsymbol{x})} \right)}{\sum_{i=1}^n K\left(\frac{d(\boldsymbol{x}, \boldsymbol{x}_i)}{h_k(\boldsymbol{x})} \right)} \tag{21}$$

where h_k is such that $\#\{i : d(\boldsymbol{x}, \boldsymbol{x}_i) < h_k\} = k$. Thus, we have to find $k_{Loss} = arg \min_{k \in \mathcal{K}} Loss(k)$. From now on, we consider $\widehat{p}_{g,k}$ the estimator of $\widehat{p}_g$.

If we use the cross-validation procedure proposed in [8] and choose as loss function

$$Loss(k) = LCV(k, i_0) = \sum_{g=1}^G \left(\mathbb{1}_{y_{i_0}=g} - p_{g,k}^{(-i_0)}(\boldsymbol{x}_{i_0}) \right)^2, \tag{22}$$

where

$$p_{g,k}^{(-i_0)}(\boldsymbol{x}_{i_0}) = \frac{\sum\limits_{i\in\mathcal{J},\, i\neq i_0} K\left(\frac{d(\boldsymbol{x}_{i_0},\boldsymbol{x}_i)}{h_k(\boldsymbol{x}_{i_0})}\right)}{\sum\limits_{i=1,\, i\neq i_0}^{n} K\left(\frac{d(\boldsymbol{x}_{i_0},\boldsymbol{x}_i)}{h_k(\boldsymbol{x}_{i_0})}\right)},$$

and $\boldsymbol{x}_{i_0}$ is the nearest neighbour of $\boldsymbol{x}$, so we denote $i_0 = arg\min\limits_{i=1,\ldots,n} d(\boldsymbol{x},\boldsymbol{x}_i)$; hence, the local choice is:

$$k_{LCV}(\boldsymbol{x}_{i_0}) = arg\min_k LCV(k,i_0) \quad (23)$$

$$k_{LCV}(\boldsymbol{x}_{i_0}) \longrightarrow h_k = h_{LCV}(\boldsymbol{x}_{i_0}) \quad (24)$$

$$Miss.\,Rate = \frac{\sum\limits_{i=1}^{n} \mathbb{1}_{y_i\neq y_i^{LCV}}}{n} \quad (25)$$

III. NONPARAMETRIC CLASSIFICATION OF UNCORRELATED SURFACES

Let us consider

$$\boldsymbol{\psi} = \{\boldsymbol{\psi}(x_1,\ldots,x_n);(x_1,\ldots,x_n)\in\mathbb{R}^n\}$$

a random n-dimensional supported f.r.v. The observed realization ψ of $\boldsymbol{\psi}$ is referred a n-dimensional f.d. In the particular case of $n = 2$, that is, of $\mathbb{R}^2$, a regular grid is chosen with nodes having coordinates $((x_1,y_1),\ldots,(x_N,y_M))$. Hence, in the following, we refer to an $M \times N$ rectangular regular grid.

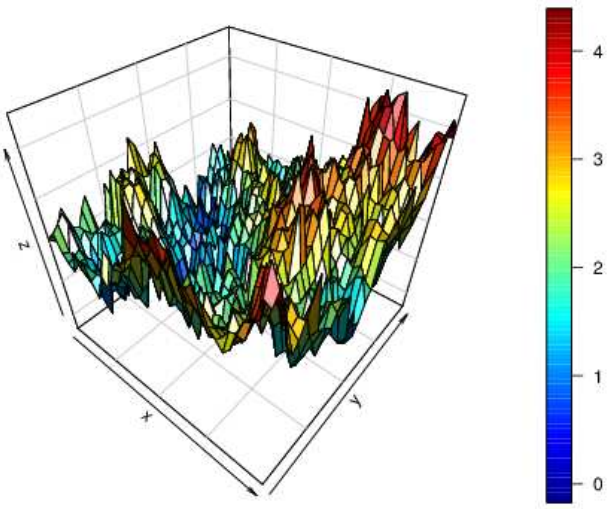

Figure 1: Discretized surface of Class 1 with $G = 2$ in our simulation study

III..I. REFORMULATION OF SEMI–METRICS

The corresponding reformulation of semi-metric based on FPCA is straightforward. In particular, when $n = 2$, we have

$$d_q^{FPCA}(\psi_1,\psi_2) = \sqrt{\sum_{j=1}^{q}\left(\sum_{i=1}^{I} w_i x_i^* v_{ji}\right)^2}, \quad (26)$$

where $1 \leq q \leq n$ the number of components chosen, $\boldsymbol{\Sigma}_{\boldsymbol{\chi}}(s,t) = \frac{1}{n}\sum\limits_{i=1}^{n}\boldsymbol{\chi}_i(s)\boldsymbol{\chi}_i(t)$ is the empirical version of the covariance kernel, $\boldsymbol{v}_j$, $j = 1,\ldots,q$, are the orthonormal eigenvectors (corresponding to the components chosen) of empirical covariance matrix $\mathbf{W}^{1/2}\boldsymbol{\Sigma}\mathbf{W}^{1/2}_{I\times I}$, with $\boldsymbol{W} = diag(w_1,\ldots,w_I)$ whose diagonal entries are two-dimensional quadrature weights, and $(x_1^*,\ldots,x_I^*) =$

$$= ((\psi_1-\psi_2)(x_1,y_1),\ldots,(\psi_1-\psi_2)(x_N,y_M)) \quad (27)$$

remains being a real vector, with $I = N \times M$. As previously, $(x_i,y_j,\ i=1,\ldots,N;\ j=1,\ldots,M) \in D \subset \mathbb{R}^2$ represents the set of nodes of a regular rectangular grid, with associated discretized functional value of the observed f.d. given by $(\psi(x_1,y_1),\ldots,\psi(x_N,y_M))$, which can also be treated as a real vector associated with the discrete observation of $\boldsymbol{\psi}$.

Reformulation of FPLSR in the two-dimensional case can be derived in a similar way. Thus,

$$d_q^{FPLSR}(\psi_1,\psi_2) = \sqrt{\sum_{j=1}^{g}\left(\sum_{i=1}^{I} w_i x_i^* v_{ji}^q\right)^2} \quad (28)$$

where x_i^*, $i = 1,\ldots,I$, are given as in equation (27), and $\boldsymbol{v}_1^q,\ldots,\boldsymbol{v}_g^q$ are performed by FPLSR.

Although it is out of our scope, semi-metrics based on derivatives can be also reformulated by considering the corresponding L^2 norm of the corresponding partial derivatives. In particular, for $n = 2$, non-uniform rational B-spline (NURBS) can be used (see, e.g., [37, 38]).

III..II. SMOLYAK QUADRATURE

We will describe the n-dimensional version of Smolyak quadrature rule to obtain a set of weights, defining, in particular, the metric $W^{1/2}\Sigma W^{1/2}_{I\times I}$, in the numerical approximation of the integral by a weighted sum of values of the integrand at certain nodes (see, e.g., [11, 20]).

The main goal is to approximate

$$I_W^n f := \int_{\prod_{i=1}^n I_i} f(x_1,\ldots,x_n)\prod_{i=1}^{n} W_i(x_i)\,dx_i$$

by a n–sequence of k_{l_j}-point quadratures ($k_{l_j} = 2^{l_j-1}+1$), where $j \in \{1,\ldots,n\}$:

$$U_{l_j} := \sum_{i=1}^{k_{l_j}} w_i f(x_i) = \sum_{i=1}^{2^{l_j-1}+1} w_i f(x_i) \quad (29)$$

with $l_j \geq 1$. Smolyak rule combines, by means of tensor products, univariate quadratures rules U_{l_j}, $j =$

$1, \ldots, n$, respectively associated with each dimension j, for $j = 1, \ldots, n$ (e.g., Trapezoidal rule, Clenshaw-Curtis rule, Gauss-Legendre rule, Gauss-Patterson rule, etc).

Definition 3 *Let $S : \mathcal{C}(\Omega) \to \mathbb{R}$ and $T : \mathcal{C}(\Xi) \to \mathbb{R}$ be operators that admit a representation of the form:*

$$\begin{aligned} Sf(\boldsymbol{x}) &= \sum_{i=1}^{m} a_i f(x_i) \\ Tg(\boldsymbol{y}) &= \sum_{j=1}^{n} b_j g(y_j) \end{aligned} \tag{30}$$

with positive weights, $\boldsymbol{x} = (x_1, \ldots, x_m)$ and $\boldsymbol{y} = (y_1, \ldots, y_n)$. The tensor product of S and T is the linear operator $S \otimes T : \mathcal{C}(\Omega \times \Xi) \to \mathbb{R}$ defined by:

$$Sf \otimes Tg(\boldsymbol{x}; \boldsymbol{y}) = \sum_{i=1}^{m} \sum_{j=1}^{n} a_i b_j f(x_i) g(y_j). \tag{31}$$

Let $\left(U_{l_j}^{(j)}\right)_{j=1}^{n}$ be a sequence of univariate quadrature rules, where j represents the dimension in which we are integrating and $k_{l_j} = 2^{l_j - 1} + 1$ the number of evaluation points. This univariate rules are chosen in such a way such that $I_{W_j}^1 p = U_{l_j}^{(j)}$, where p is a polynomial of degree at most k_{l_j}.

We denote as $\left(w_i^{(j)}\right)_{i=1}^{k_{l_j}}$ and $\left(x_i^{(j)}\right)_{i=1}^{k_{l_j}}$ the weights and the nodes (resp.) of the univariate rule $U_{l_j}^{(j)}$, for $j = 1, \ldots, n$. Thus, the original problem can be approximated in tensor product form:

$$I_W^n f \approx \bigotimes_{j=1}^{n} U_{l_j}^{(j)} f = Q_k^n \tag{32}$$

with $\# \left\{U_{l_j}^{(j)}\right\} = k_{l_j} = 2^{l_j - 1} + 1$ and $\boldsymbol{l} = (l_1, \ldots, l_n)$, with $l_j \leq k$, for all $j \in \{1, \ldots, n\}$.

In fact, Smolyak quadrature rule proposed in [20] and [12] uses difference operators instead of directly applying the tensor product.

Definition 4 *Let $\left(U_i^{(j)}\right)_{i=1}^{\infty}$ be a sequence of univariate rules in I_j. We define the difference operators in I_j as:*

$$\Delta_0^{(j)} = 0,\ \Delta_1^{(j)} = U_1^{(j)} \text{ and } \Delta_{i+1}^{(j)} = U_{i+1}^{(j)} - U_i^{(j)} \tag{33}$$

Thus, Smolyak quadrature rule of order k in the n-dimensional rectangle $I_1 \times \ldots \times I_n$ (for simplicity we assume $I^n = I \times \ldots \times I$) can be defined as the operator:

$$Q_k^n = \sum_{\|\boldsymbol{\alpha}\|_1 \leq k} \bigotimes_{j=1}^{n} \Delta_{\alpha_j}^{(j)} \tag{34}$$

where $\boldsymbol{\alpha} \in \mathbb{N}^n$ and $\alpha_j > 0$ (which implies that $k \geq n$). Remark that in the case of $n = 1$, $Q_k^1 = U_k^{(1)}$.

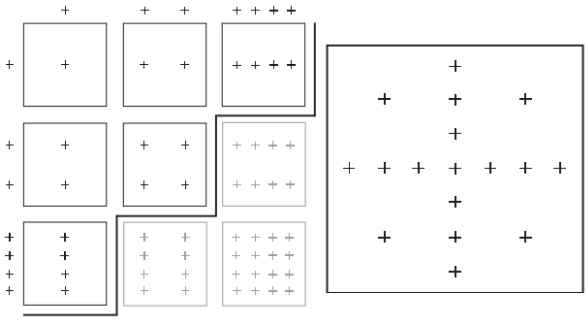

Figure 2: Product grids $X_{i_1}^{(1)} \times X_{i_2}^{(1)}$ such that $\|(i_1, i_2)\|_\infty \leq 3$ (on left) and the Q_4^2 grid (on right)

Since there are many terms that are canceled, we shall also present a combination method of Smolyak rules (see, e.g., [42]):

$$Q_k^n = \sum_{\substack{m \leq \|\boldsymbol{\alpha}\|_1 \leq k \\ \boldsymbol{\alpha} \in \mathbb{N}^n \\ \boldsymbol{\alpha} \geq \mathbf{1}}} (-1)^{k - \|\boldsymbol{\alpha}\|_1} \binom{n-1}{k - \|\boldsymbol{\alpha}\|_1} \bigotimes_{j=1}^{n} U_{\alpha_j}^{(j)} \tag{35}$$

with $m = max\{n, k - n + 1\}$.

Rewritting (34) and using (32), we obtain:

$$Q_k^n = \sum_{l=m}^{k} \sum_{\substack{\|\boldsymbol{\alpha}\|_1 = l \\ \boldsymbol{\alpha} \in \mathbb{N}^n \\ \boldsymbol{\alpha} \geq \mathbf{1}}} \sum_{j_1=1}^{k_{\alpha_1}} \cdots \sum_{j_n=1}^{k_{\alpha_n}} c(k, n, l) w_{j,\boldsymbol{\alpha}} f(\boldsymbol{x}_{j,\boldsymbol{\alpha}}) \tag{36}$$

where $c(k, n, l) = (-1)^{k-l} \binom{n-1}{k-l}$, $w_{j,\boldsymbol{\alpha}} = w_{j_1}^{(\alpha_1)} \ldots w_{j_n}^{(\alpha_n)}$ and $\boldsymbol{x}_{j,\boldsymbol{\alpha}} = \left(x_{j_1}^{(\alpha_1)} \ldots x_{j_n}^{(\alpha_n)}\right)$.

III..I. NUMERICAL IMPLEMENTATION

The main steps and auxiliary functions in the implementation of Smolyak quadrature are the following :

- **Step 1** Define the function that provides us univariate nodes and weights (univariate quadrature rules at each dimension).
- **Step 2** Generate all multi-indices satisfying restrictions established in the algorithm proposed in [12]. For instance, if $n = 3$ and $k = 5$, $\boldsymbol{\alpha}$ could be $(1,1,1)$, $(1,1,2)$, $(1,2,1)$, $(2,1,1)$, $(1,1,3)$, $(1,3,1)$, $(3,1,1)$, $(1,2,2)$, $(2,1,2)$ and $(2,2,1)$.
- **Step 3** Determine, for any vector sequence $\left(\boldsymbol{v}^{(i)}\right)_{i=1}^{l}$, with $\boldsymbol{v}^{(i)} \in \mathbb{R}^{n_i}$, $i = 1, \ldots, l$, its vector combination. Thus, we define inductively $c_{\boldsymbol{v},l} = \text{combvec}\left(\left(\boldsymbol{v}^{(i)}\right)_{i=1}^{l}\right)$ as follows:

$$c_{\boldsymbol{v},l} = \begin{pmatrix} c_{\boldsymbol{v},l-1} \ldots c_{\boldsymbol{v},l-1} \ldots c_{\boldsymbol{v},l-1} \ldots c_{\boldsymbol{v},l-1} \\ v_1^{(l)} \underset{(n_l-1)}{\ldots} v_1^{(l)} \ldots v_{n_l}^{(l)} \underset{(n_l-1)}{\ldots} v_{n_l}^{(l)} \end{pmatrix} \tag{37}$$

with $c_{\boldsymbol{v},1} = \boldsymbol{v}_{(1)} = (v_1^{(1)} \dots v_{n_1}^{(1)})$.

In addition, we have implemented two more functions. A function that groups weights associated at the same node, and auxiliary function that deletes the nodes with total weight equal to zero. Smolyak-nodes are different from the nodes where we have our observations, so we previously interpolate our f.dat. considering locally polynomials or k-Nearest Neighborhood Smoother. The assignment of weights is done in two ways: To each interpolated node, we assign the weight corresponding to the Nearest Neighbour Smolyak node; or, alternatively, we assign the weight defined by the average of the weights associated with the $k_{Smolyak}$-Nearest Neighborhood Smolyak nodes.

IV. FUNCTIONAL CLASSIFICATION RESULTS OF CURVES

The performances of the proposed nonparametric curve classification methodology, as well as of the one formulated in [8] is now compared in terms of their implementation from a spectrometric curve dataset available at url http://lib.stat.cmu.edu/datasets/tecator. This dataset is related to quality control in food industry. It corresponds to a sample of finely chopped meat. For each unit i, among 215 pieces, we observe one spectrometric curve (χ_i f.d.) which corresponds to the absorbance measured at 100 wavelengths. Moreover, we have measured its fat content y_i, for $i = 1, \dots, 215$, obtained by an analytical chemical processing.

In the implementation of the classification procedure for validation purposes, our f.dat. sample has been randomly splitted into two sub-samples respectively corresponding to the training f.dat. sample, which constitutes a 70% of the total dataset, and a f.dat. validation sample or test sample, which in our case constitutes a 30% of the total sample.

Figure 3 shows spectrometric f.dat. The magnitude plotted is absorbance versus wavelength for different pieces, where 100 channel spectrum of absorbances are showed. Hence, each data appears as a discretized curve in 100 points, and interpolation is performed to get the corresponding values in a finer partition of the set containing the 100 points within the same wavelength interval $850 - 1050$ (see Figure 5). Two catgories or groups are distinguished in advance: fat content under 20 ($y_i = 1$) and over 20 ($y_i = 2$) (see Figure 4).

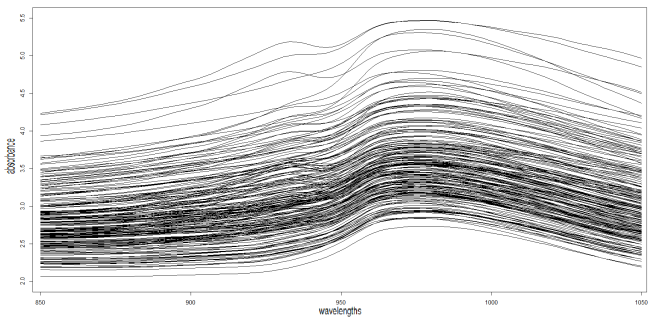

Figure 3: Discretized spectometric curves.

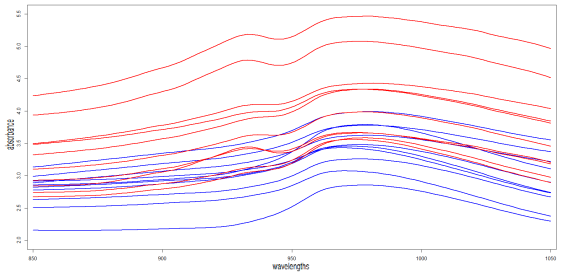

Figure 4: Discretized curves splitted by groups: the blue ones belong to class 1 (low fat content) and the red ones belong to class 2 (higher fat content).

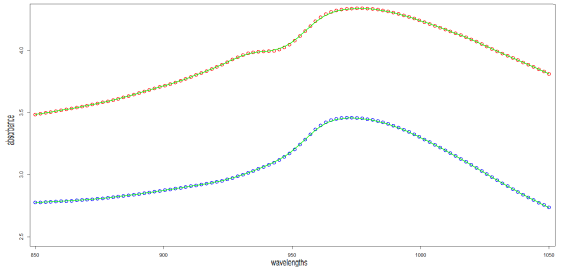

Figure 5: Accuracy of interpolation of a curve at each category, with step called $step_{mesh}$.

Figure 6 shows the results obtained using FPCA semi-metric, when different kernels (quadratic, indicator and triangle) and inputs (components, factors or orders) are considered, using the methodology given in [8], by means of 50 simulations.

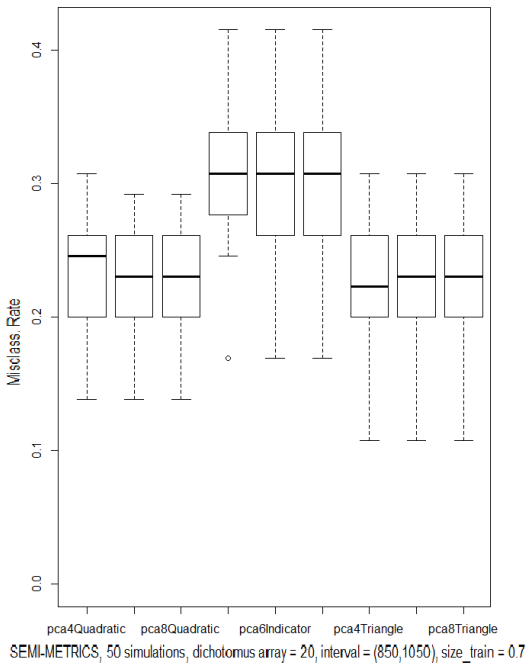

Figure 6: Misclassification rate of functional classification using the method proposed in [8], with FPCA semi-metric.

Figure 7 shows the results obtained using FPLSR semi-metric, when different kernels and inputs are considered, using the methodology given in [8].

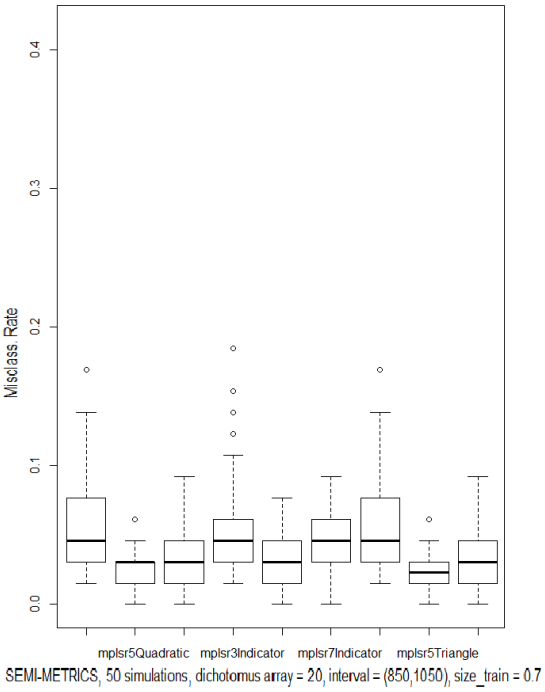

Figure 7: Misclassification rate of functional classification using the method proposed in [8], with FPLSR semi-metric.

Figure 8 shows the results obtained using a semi-metric based on derivatives, when different kernels and inputs are considered.

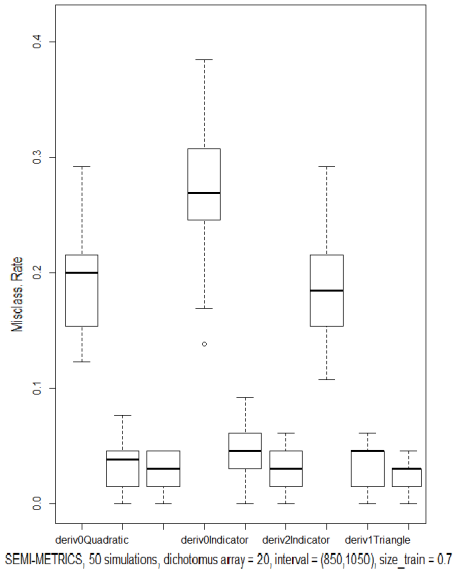

Figure 8: Misclassification rate of functional classification using the method proposed in [8], with semi-metric based on derivatives.

At each one of these box-plots, we reflect results obtained with implementation of a quadratic kernel in the first three ones, the next three ones reflect results with indicator kernel, and the three last ones show the results with triangle kernel. Alternatively, Figures 9, 10 and 11 display the results using our methodology in terms of Smolyak quadrature rule considering three neighbors, implementing trapezoid rule with $k = 5$ and using discretization step equal to 0.25.

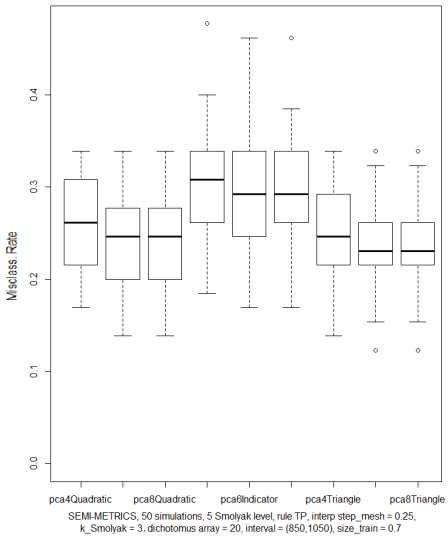

Figure 9: Results obtained with our implementation using Trapezoid rule (level 5), with discretization step equal to 0.25 and 3 neighbors, with FPCA.

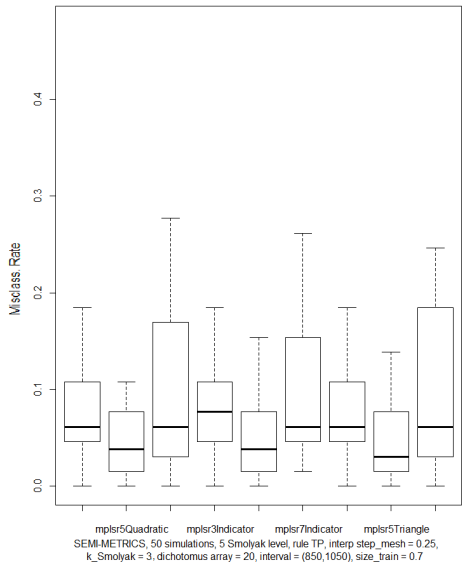

Figure 10: Results obtained with our implementation using Trapezoid rule (level 5), with discretization step equal to 0.25 and 3 neighbors, with FPLSR.

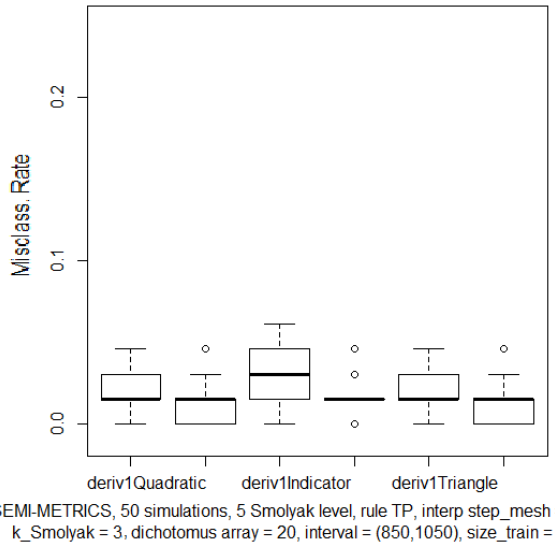

Figure 11: Results obtained with our implementation using Trapezoid rule (level 5), with discretization step equal to 0.25 and 3 neighbors, with semi-metric based on derivatives.

The following Figures 12, 13 and 14 show different implementations of our methodology with different inputs such as Clenshaw-Curtis quadrature rule or doing directly the assignment of Smolyak weights. A similar performance is obtained in comparison with the previous results displayed. One can observe that our methodology is more flexible than the one presented in [8]. However, our methodology is also affected by the interpolation error, and the error associated with the rule considered for the assigning of weights. This fact can also be observed in Figures 15 and 16, where we have used a greater interpolation step. Note that a slight improvement in the accuracy can be appreciated. Summarizing, we have to look for a compromise between precision in the numerical approximation of the integral, increasing the number of points in the sample by interpolation, and the associated interpolation and weight allocation errors.

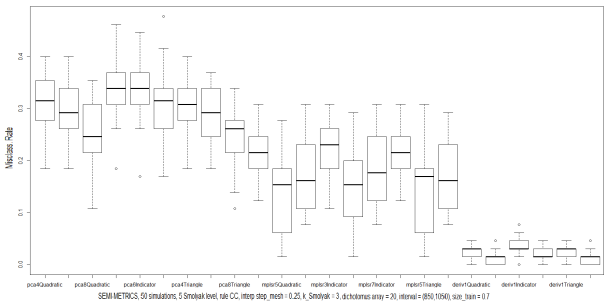

Figure 12: Results obtained with our implementation using Clenshaw-Curtis rule (level 5), with discretization step equal to 0.25 and 3 neighbours.

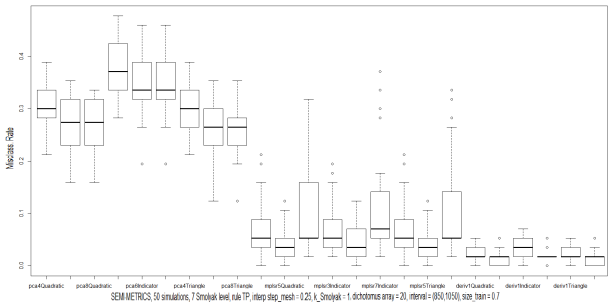

Figure 13: Results obtained with our implementation using Trapezoidal rule (level 7), with discretization step equal to 0.25.

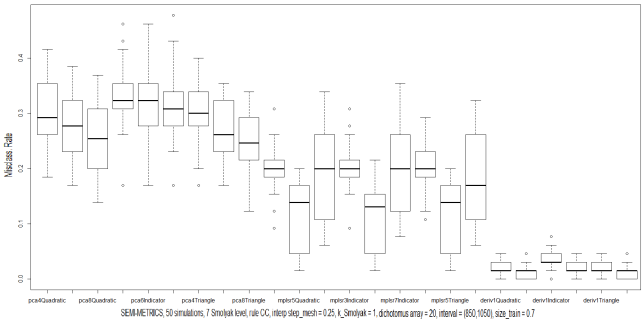

Figure 14: Results obtained with our implementation using Clenshaw-Curtis rule (level 7), with discretization step equal to 0.25.

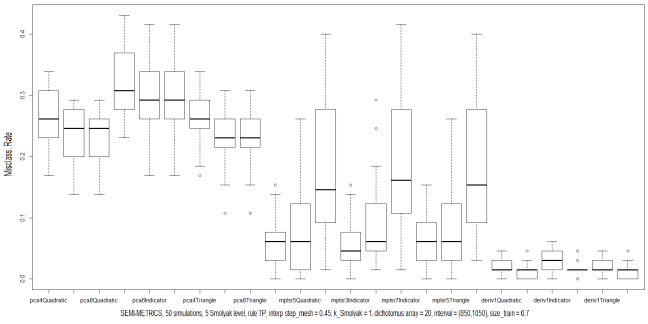

Figure 15: Results obtained with our implementation using Trapezoidal rule (level 5), with discretization step equal to 0.45.

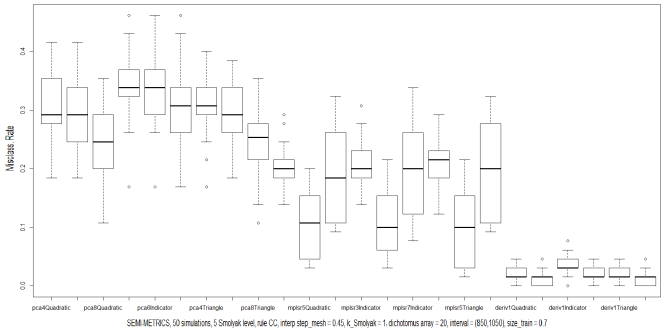

Figure 16: Results obtained with our implementation using Clenshaw-Curtis rule (level 5), with discretization step equal to 0.45.

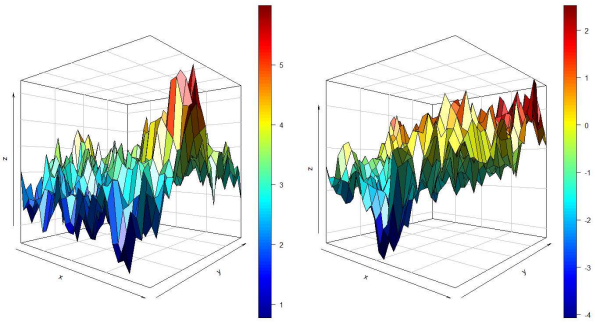

Figure 17: Surfaces plotted: on left surface belongs to category 1, on right surface belongs to category 2.

V. NUMERICAL EXAMPLE FOR FUNCTIONAL CLASSIFICATION OF TREND IN RANDOM GAUSSIAN SURFACES

A sample of 200 Gaussian random surfaces is generated, over a regular grid within the square $[1,5] \times [1,5]$, with the same integral covariance operator defined by the isotropic Gaussian kernel in two dimensions. These Gaussian surfaces have two different (which lead to the definition of our two groups), but very close, functional means (see Figure 17). Our problem consists of discrimination between different trends defining the mean value of Gaussian surfaces. This numerical example is considered previously to our main simulation study developed in the next section where, among other subjects, we solve the problem of discrimination between different spatial correlation functions characterizing the infinite-dimensional distribution of zero-mean Gaussian random surfaces.

Let us then consider the following two groups of Gaussian random surfaces:

$$\boldsymbol{\chi}^1 \sim N\left(\boldsymbol{\mu}_1 = (h_i)_{i=1,\ldots,k}, \boldsymbol{\Sigma} = \boldsymbol{C}\boldsymbol{C}^T\right) \quad (38)$$

$$\boldsymbol{\chi}^2 \sim N\left(\boldsymbol{\mu}_2 = \boldsymbol{\mu}_1 + \boldsymbol{v}, \boldsymbol{\Sigma} = \boldsymbol{C}\boldsymbol{C}^T\right) \quad (39)$$

where $h_i = \frac{i}{N \times M} + 2$, $i = 1, \ldots, k$, $k = N \times M$, with $\boldsymbol{v} = (0.5, 0.5, \ldots, 0.5) \in \mathbb{R}^{N \times M}$, and $C_{ij} = e^{-\|(x_i, y_i) - (x_j, y_j)\|_2^2}$, where $\boldsymbol{\chi}^j$ denotes a random surface of type j, $j = 1, 2$, with values defined over a regular grid given by $((x_1, y_1), \ldots, (x_N, y_M))$. We have imposed a minimum number of surfaces belonging to each class.

Note that now we have not to interpolate since we can generate surfaces as finely as we want. A minimum number of surfaces belonging to each class is fixed to ensure the representativeness of the groups. As commented, in the previous implementation of our methodology in terms of curves, we restrict our attention to the FPCA and FPLSR semi-metrics. Figures 18 and 19 then display the derived classification results, reflecting a good performance of our methodology for discrimination between different trends of Gaussian surfaces, keeping in mind that the two categories distinguished are very close.

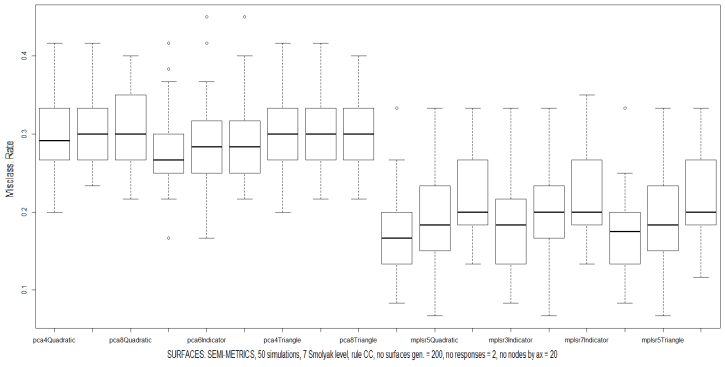

Figure 18: Results obtained with our classification methodology for surfaces using Clenshaw-Curtis rule(level 7) on a 20×20 spatial regular grid.

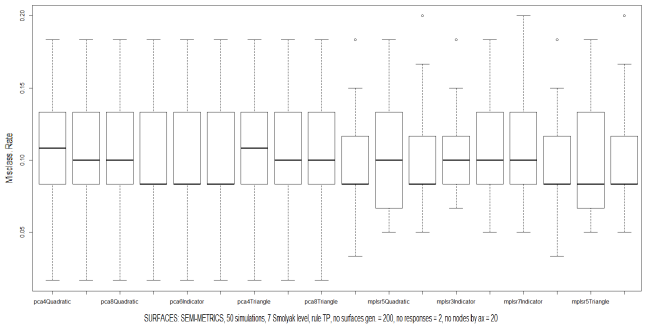

Figure 19: Results obtained with our classification methodology for surfaces using Trapezoidal rule (level 7) on a 20×20 spatial regular grid.

Consider now two groups respectively based on a linear and a non-linear, cosine type, trend surfaces:

$$\boldsymbol{\mu}_2 = \cos\left(\mu_1 \frac{\pi}{2}\right)_{i=1,\ldots,N\times M} \tag{40}$$

where $\boldsymbol{\mu}_1$ is given as before. For a lower resolution level, namely for a 12×12 regular grid, FPLSR clearly outperforms FPCA (see Figure 20). Thus, FPLS is more suitable for well-differentiated groups when numerical integration must be performed from a low quality discrete version of our surface dataset.

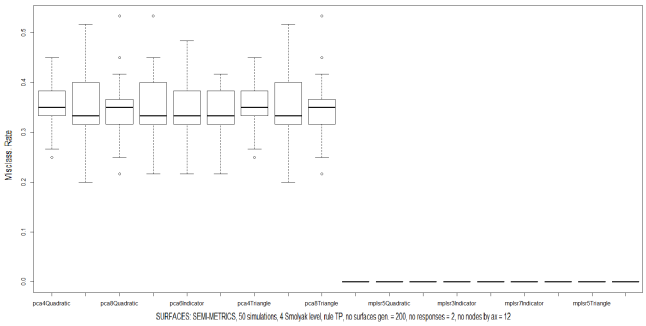

Figure 20: Results obtained with our classification methodology for surfaces using Trapezoidal rule (level 4) on a 12×12 spatial regular grid.

VI. FUNCTIONAL CLASSIFICATION RESULTS OF RANDOM AND NON-RANDOM SURFACE IRREGULARITIES OF RAILWAY TRACK

The problem of deterministic and random vibration classification from the observation of surface irregularities of railway track will be addressed in this section, which constitutes a key problem in the field of railway engineering. As commented in [25], it is very important to modeling these irregularities since the created loads resulting from them cause fatigue in the vehicles and rail beams. According to [47], the rail irregularities are the second leading cause of bridge vibrations, and the first one of train vibrations.

Two types of rail irregularities are studied in [25, 47]: random and non-random irregularities. Random irregularities include the roughness of the rails. Here, these irregularities are represented in terms of zero-mean Gaussian surfaces with different spatial functional correlations. Deterministic irregularities are usually represented in terms of a irregularity function of the railway $r(x)$ (see, for example, [9]).

VI..I. NON-RANDOM SURFACES IRREGULARITIES

As proposed in [25, 47] approach, an one-dimensional railway track is firstly considered. We can see in the example shown in Figure 21 that a simple beam of span length $L = 50\ m$ is analyzed. We denote as B the distance from the origin to the first irregularity, and A the constant rail length between two imperfections (see Figure 21).

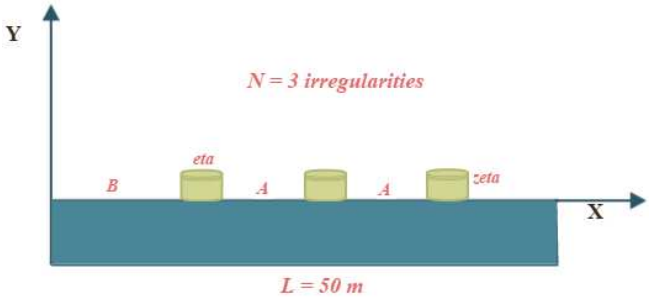

Figure 21: Illustrative and very simple example of the approach, with $L = 50\ m$ and $N = 3$.

Setting the number of these irregularities in the railway track of length L, denoted as N, and considering the depth and the length of the imperfections (ζ and η respectively, as shown in Figure 21), we can establish the following formula:

$$N = \frac{L - B}{A + \eta} \tag{41}$$

Let us now consider three different models of imperfections:

- Model 1: $N_1 = 3$ and $B_1 = 4.5$.
- Model 2: $N_2 = 4$ and $B_2 = 2.5$.
- Model 3: $N_3 = 5$ and $B_3 = 1$.

We divide each one of them into two models using different values of A:

- Model 1: $N_1 = 3$, $B_1 = 4.5$ and $A_1 = 3.5$.
- Model 2: $N_2 = 3$, $B_2 = 2.5$ and $A_2 = 5.2$.
- Model 3: $N_3 = 4$, $B_3 = 1$ and $A_3 = 3.5$.
- Model 4: $N_4 = 4$, $B_4 = 4.5$ and $A_4 = 5.2$.
- Model 5: $N_5 = 5$, $B_5 = 2.5$ and $A_5 = 3.5$.
- Model 6: $N_6 = 5$, $B_6 = 1$ and $A_6 = 5.2$.

Using two different values of ζ, the final set of models is given in Table 1:

Table 1: Final models

Models	N	B (m)	A (m)	ζ (m)
Model 1	3	4.5	3.5	0.007
Model 2	3	4.5	5.2	0.007
Model 3	3	4.5	3.5	0.015
Model 4	3	4.5	5.2	0.015
Model 5	4	2.5	3.5	0.007
Model 6	4	2.5	5.2	0.007
Model 7	4	2.5	3.5	0.015
Model 8	4	2.5	5.2	0.015
Model 9	5	1	3.5	0.007
Model 10	5	1	5.2	0.007
Model 11	5	1	3.5	0.015
Model 12	5	1	5.2	0.015

Choosing any of them, and using formula (41), we can get the corresponding set of values of η (see Table 2):

Table 2: Set of values of η

Models	η (m)
Model 1	11.667
Model 2	9.967
Model 3	11.667
Model 4	9.967
Model 5	8.375
Model 6	6.675
Model 7	8.375
Model 8	6.675
Model 9	6.300
Model 10	4.600
Model 11	6.300
Model 12	4.600

As proposed in [9], for each one of these models, denoted as M_i $(i = 1, \ldots, 12)$, the non-random irregularities can be mathematically defined by the following function:

$$r(x) = \begin{cases} \frac{\zeta}{2}\left(1 - \cos\left(\frac{2\pi x}{\eta}\right)\right) & \text{if } C \leq x \leq C + \eta \\ 0 & \text{elsewhere} \end{cases} \tag{42}$$

where $C = B + k(A + \eta)$, $k = 0, 1, \ldots, N$.

As we want to deal with surfaces, in this paper we shall extend this approach to the two-dimensional framework. Such as the rail width is quite smaller than L, we use a anisotropic model where the imperfections are deployed through the 'x' axis (see Figures 22).

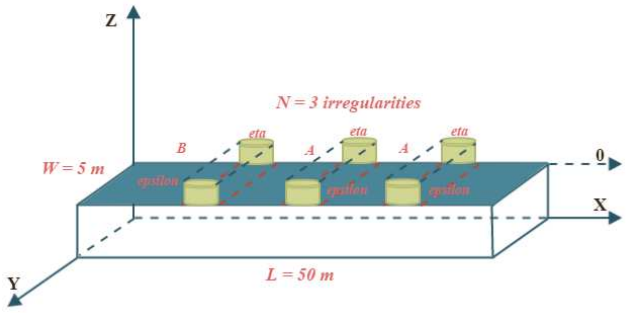

Figure 22: Illustrative and very simple example of the two-dimensional approach, with $L = 50\ m$, $W = 2.5\ m$ and $N = 3$

Extending equation (42) and setting $W = 2.5$, we have

$$r(x, y) = \begin{cases} \frac{\zeta}{2}\left(1 - \cos\left(\frac{2\pi x}{\eta}\right)\right) & \text{if } C \leq x \leq C + \eta \\ 0 & \text{elsewhere} \end{cases} \tag{43}$$

where $C = B + k(A + \eta)$, $k = 0, 1, \ldots, N$ and $y \in [0, W]$.

As well as in previous sections we have been working with a square regular grid, where the length coincides with the width, a rectangular grid, with $L = 50\ m$ and $W = 2.5\ m$ is used now. In section III..II., for simplicity we have assumed $I^n = I \times \cdots \times I$, but this implementation used in the previous section is not valid here, and we have to recalculate all the steps of the proposed numerical integration algorithm for functional classification of noisy Gaussian surfaces. We have then obtained from formula (36):

$$Q_k^n = \sum_{l=m}^{k} \sum_{\substack{\|\boldsymbol{\alpha}\|_1 = l \\ \boldsymbol{\alpha} \in \mathbb{N}^n \\ \boldsymbol{\alpha} \geq \mathbf{1}}} \sum_{j_1=1}^{k_{\alpha_1}} \cdots \sum_{j_n=1}^{k_{\alpha_n}} c(k, n, l) w_{j,\boldsymbol{\alpha}} f(\boldsymbol{x}_{j,\boldsymbol{\alpha}}) \tag{44}$$

where $c(k, n, l) = (-1)^{k-l}\binom{n-1}{k-l}$, $w_{j,\boldsymbol{\alpha}} = w_{j_1}^{(\alpha_1)} \ldots w_{j_n}^{(\alpha_n)}$ and $\boldsymbol{x}_{j,\boldsymbol{\alpha}} = \left(x_{j_1}^{(\alpha_1)} \ldots x_{j_n}^{(\alpha_n)}\right)$.

Note that, in the previous section, $x_{j_i}^{(\alpha_i)} \in I$, for all $i = 1, \ldots, n$. However, we now compute $x_{j_i}^{(\alpha_i)}$ such as $x_{j_i}^{(\alpha_i)} \in I_i\ \forall i = 1, \ldots, n$. Rewritting (44), we obtain:

$$Q_k^{n,\boldsymbol{L}} = \sum_{l=m}^{k} \sum_{\substack{\|\boldsymbol{\alpha}\|_1 = l \\ \boldsymbol{\alpha} \in \mathbb{N}^n \\ \boldsymbol{\alpha} \geq \mathbf{1}}} \sum_{j_1=1}^{k_{\alpha_1}} \cdots \sum_{j_n=1}^{k_{\alpha_n}} c(k, n, l) w_{j,\boldsymbol{\alpha}} f(\boldsymbol{x}_{j,\boldsymbol{\alpha}}) \tag{45}$$

where $c(k, n, l) = (-1)^{k-l}\binom{n-1}{k-l}$, $w_{j,\boldsymbol{\alpha}} = w_{j_1}^{(\alpha_1)} \ldots w_{j_n}^{(\alpha_n)}$ are the weights of the $U_{l_j}^{(j)}$ univariate quadrature in I_j, $\boldsymbol{x}_{j,\boldsymbol{\alpha}} = \left(x_{j_1}^{(\alpha_1)} \ldots x_{j_n}^{(\alpha_n)}\right) \in I_1 \times \cdots \times I_n$ and $\boldsymbol{L}$ is an interval matrix where $L_i j = a_{ij}$, $i = 1, \ldots, n$, $j = 1, 2$, with $I_i = (a_{i1}, a_{i2})$.

Figures 23, 24 and 25 provide a zoom of the generated irregularity models for the two-dimensional deterministic case.

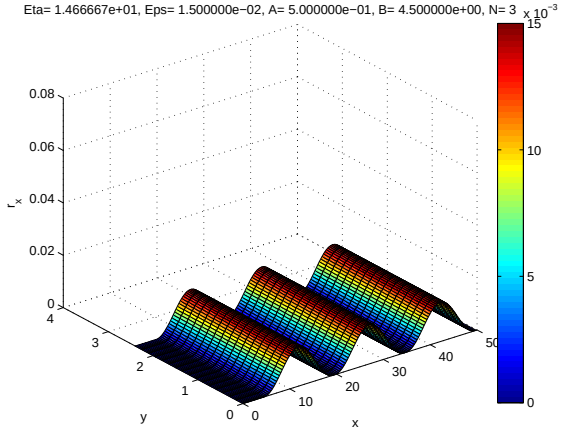

Figure 23: Irregularity belongs to model M_3.

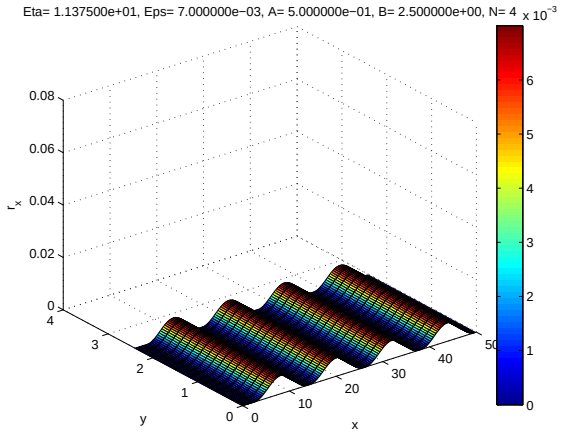

Figure 24: Irregularity belongs to model M_5.

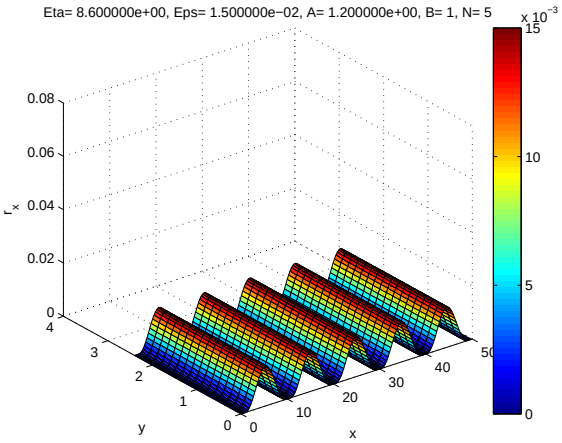

Figure 25: Irregularity belongs to model M_{12}.

It is assumed that our observed irregularities are measured by a device that introduces and additive zero-mean Gaussian noise. That is, they are perturbed by such a noise as follows:

$$S_i(x,y) = r(x,y) + \varepsilon(x,y) \tag{46}$$

for $i = 1,\ldots,12$, models considered, and for $\varepsilon(x,y) \sim N\left(\boldsymbol{\mu} = \mathbf{0}, \boldsymbol{\Sigma} = \sigma_i^2 \boldsymbol{Id}\right)$ being a Gaussian white noise with standard deviation $\sigma_i = \frac{\eta_i}{2}$, $i = 1,\ldots,12$. Figures 26, 27 and 28 show again a zoom of the perturbed Gaussian surfaces.

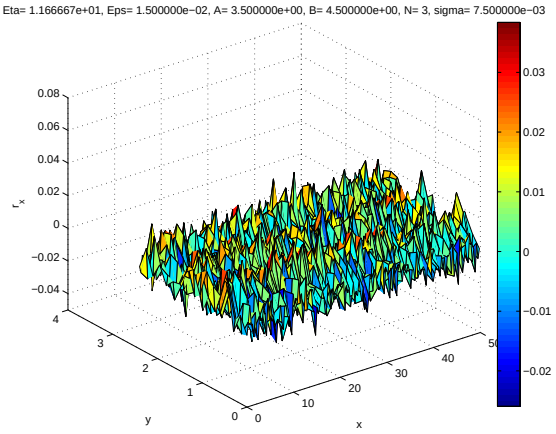

Figure 26: Irregularity perturbed belongs to model M_3.

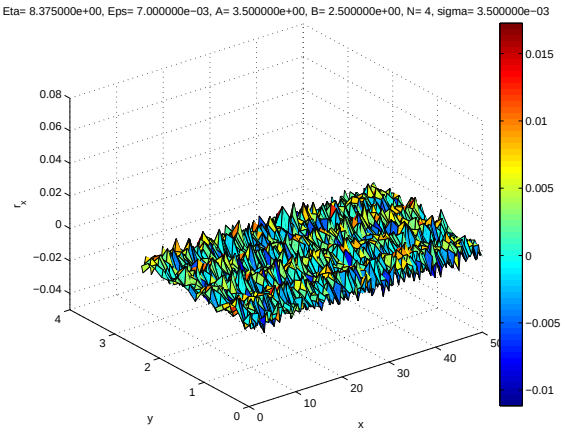

Figure 27: Irregularity perturbed belongs to model M_5.

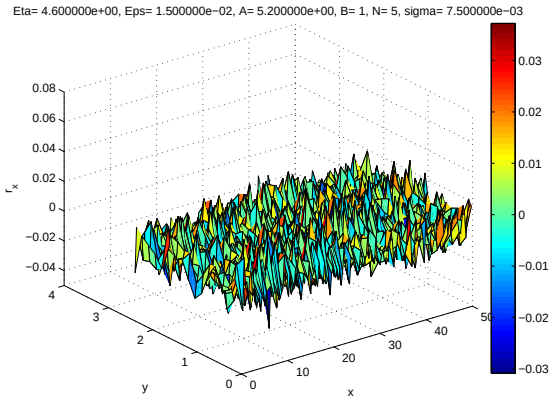

Figure 28: Irregularity perturbed belongs to model M_{12}.

We consider a regular grid corresponding to discretization steps 1.3 in length, and 0.3 in width. A minimum number of surfaces belonging to each group has been set and 50 simulations have been running. Applying the same methodology as the one used in Section V. with a sample of 500 surfaces, we obtain the results shown in Figure 29.

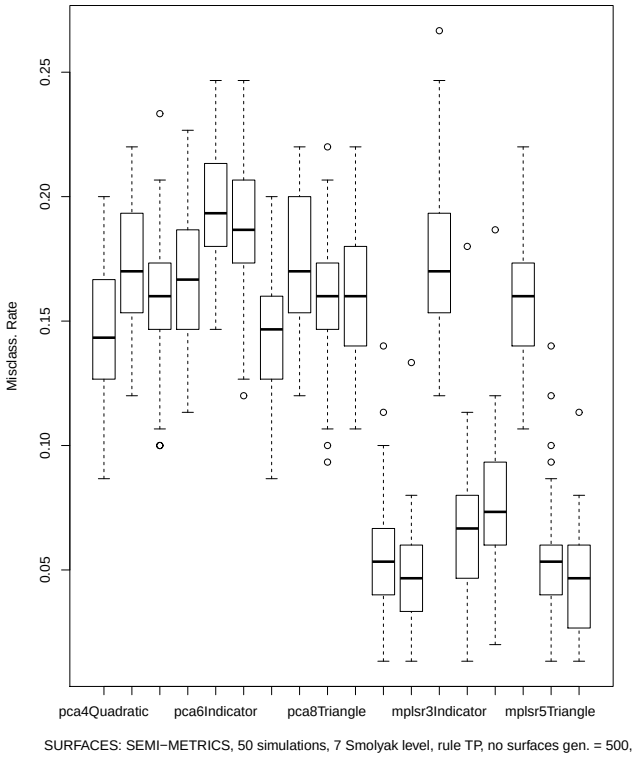

Figure 29: Results obtained with our classification methodology for non-random irregularities using Trapezoidal rule (level 7).

Remark that the accuracy depends on the magnitude of σ_i, $i = 1, \ldots, 12$, and the length of the gap between the irregularities (A). One can observe that to a greater σ_i, $i = 1, \ldots, 12$, corresponds a better performance. For the same reason, we get a better accuracy using a greater value of A (see Figure 30).

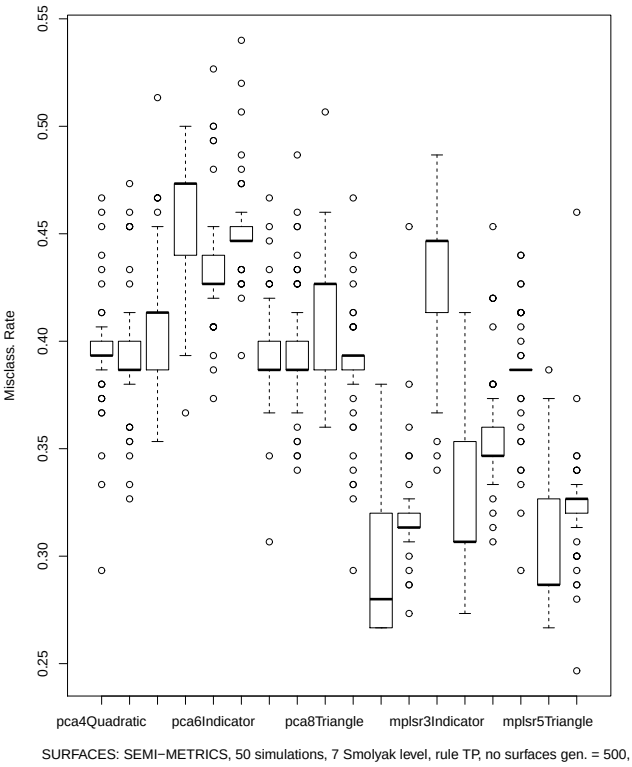

Figure 30: Results obtained with our classification methodology for non-random irregularities using Trapezoidal rule (level 7) and $\boldsymbol{A}_{new} = (1.5, 2.2)$ instead of previous $\boldsymbol{A} = (3.5, 5.2)$.

VI..II. RANDOM SURFACES IRREGULARITIES

As commented before, rail imperfections can be divided into deterministic and random imperfections. Different factors may be the cause of these random irregularities, as imperfections in material or in rail joints, errors during design, among others.

We are going to focus on the little roughness of the rails, which is included in random imperfections, by means of Gaussian surfaces. Since we will consider little roughness, distributions with null mean will be considered, taking into account that the origin of ordinate axis is represented by the rail. Generating a sample of 200 Gaussian surfaces, we will distinguish the following four categories of roughness (see Figures 31, 32, 33 and 34):

$$\boldsymbol{\chi}^h \sim N\left(\boldsymbol{\mu}_h = \mathbf{0}, \boldsymbol{\Sigma} = \boldsymbol{C}_h \boldsymbol{C}_h^T\right) \tag{47}$$

where $h = 1, 2, 3, 4$ identifies our categories,

and

$$C_{hij} = \frac{k_h}{LW} e^{\frac{-\left\|\left(\frac{x_i}{L}, \frac{y_i}{W}\right) - \left(\frac{x_j}{L}, \frac{y_j}{W}\right)\right\|_2}{k_h}} \quad (h = 1, 3)$$

represents the correlation structure model for each group $h = 1, 3$, within the family of Ornstein-Uhlenbeck covariance kernels, and

$$C_{hij} = \frac{k_h}{LW} e^{\frac{-\left\|\left(\frac{x_i}{L}, \frac{y_i}{W}\right) - \left(\frac{x_j}{L}, \frac{y_j}{W}\right)\right\|_2^2}{k_h}} \quad (h = 2, 4)$$

within the family of spatial correlations functions given by the non-linear isotropic Gaussian kernel, using a vector of scales $\boldsymbol{k}_h = (0.04, 0.04, 0.06, 0.06)$. Both correlation models correspond to weak dependence in space (see Figures 31, 32, 33 and 34). As previously, a minimum number of surfaces belonging to each class has been fixed, and the two-dimensional rectangle $[0, L] \times [0, W]$ has been considered, with discretization step size 1.3 in length, and discretization step size 0.3 in width.

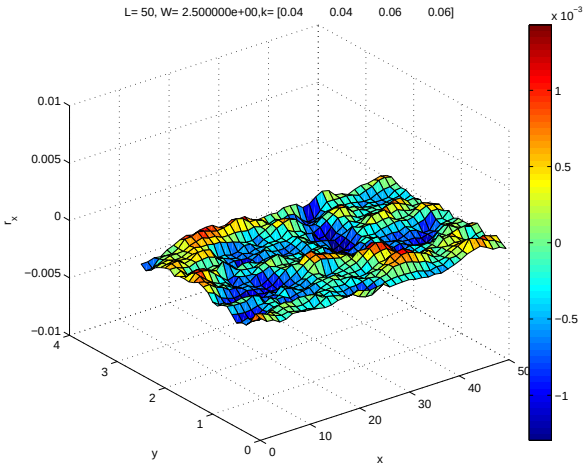

Figure 31: Random surface belongs to category 1, using the isotropic Ornstein–Uhlenbeck covariance kernel and $k_h = 0.04$ (weak correlated model).

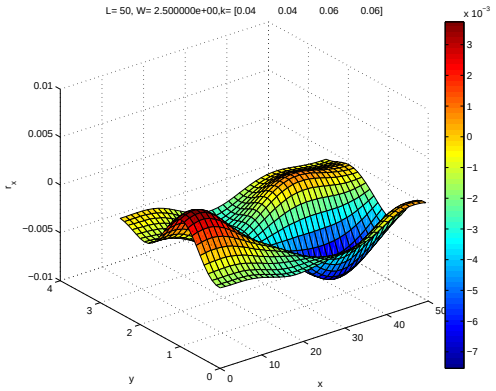

Figure 32: Random surface belongs to category 2, using the isotropic Gaussian covariance kernel and $k_h = 0.04$ (weak correlated model).

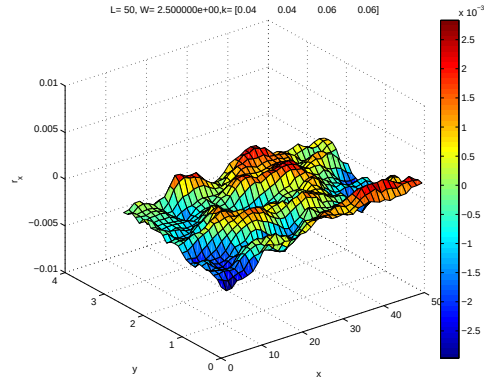

Figure 33: Random surface belongs to category 3, using the isotropic Ornstein-Uhlenbeck covariance kernel and $k_h = 0.06$ (strong correlated model).

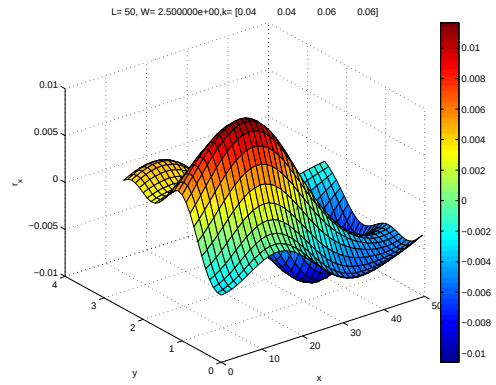

Figure 34: Random surface belongs to category 4, using the isotropic Gaussian covariance kernel and $k_h = 0.06$ (strong correlated model).

One can observe from the random surfaces displayed in

Figures 31, 32, 33 and 34, that the random surfaces with covariance matrix given by the isotropic Gaussian kernel display a smoother local behavior than the ones with Ornstein–Uhlenbeck correlation kernel. Note that the first ones display stronger spatial correlations (see equation (47)). Parameter k_h within each spatial functional correlation family represents the spatial dependence range (scale parameter) of each random surface class. Applying our methodology as in Subsection VI..I. with a sample of 200 surfaces, the following results are obtained (see Figure 35):

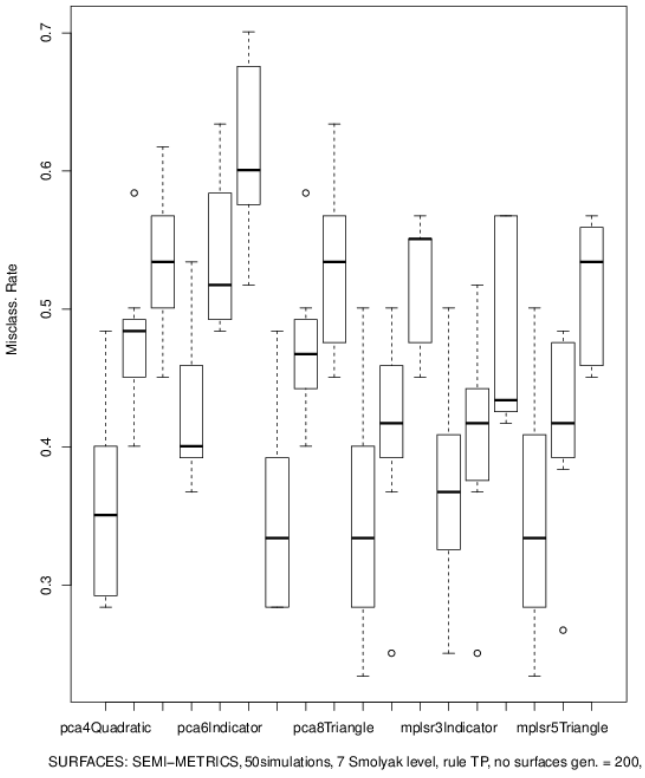

Figure 35: Results obtained with our classification methodology for random irregularities using Trapezoidal rule (level 7), using a weak correlated model.

Weak correlated surfaces, e.g., $k_h = (0.04, 0.04, 0.5, 0.5)$, are displayed in Figures 36 and 37, while Figures 38 and 39 show strong-correlated surfaces.

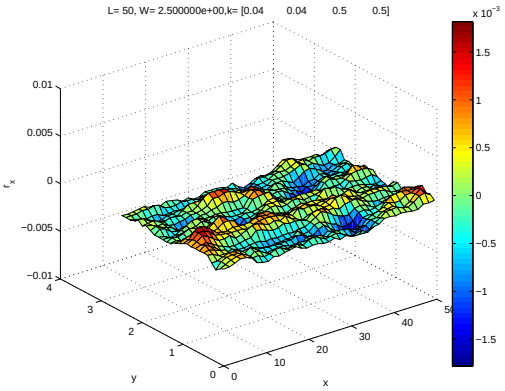

Figure 36: Random surface belongs to category 1, using the isotropic Ornstein–Uhlenbeck covariance kernel and $k_h = 0.04$ (weak spatial correlated surfaces).

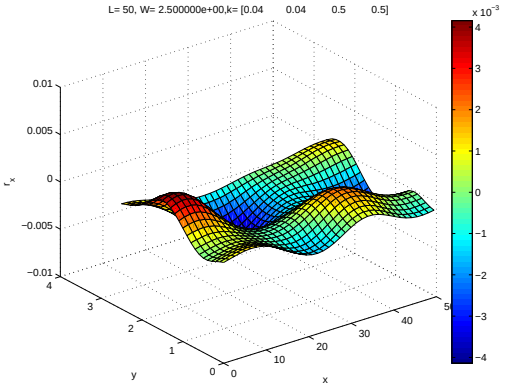

Figure 37: Random surface belongs to category 2, using the isotropic Gaussian covariance kernel and $k_h = 0.04$ (weak spatial correlated surfaces).

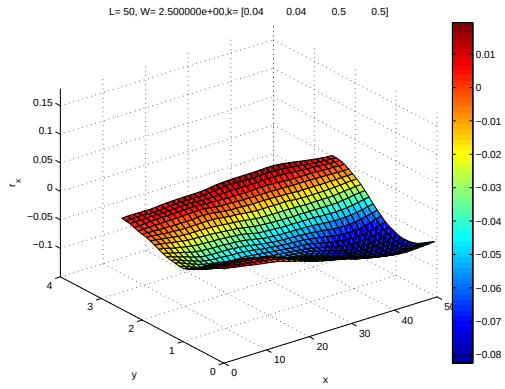

Figure 38: Random surface belongs to category 3, using the isotropic Ornstein–Uhlenbeck covariance kernel and $k_h = 0.5$ (strong spatial correlated surfaces).

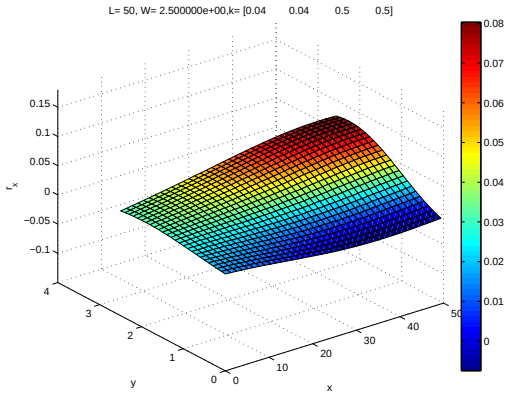

Figure 39: Random surface belongs to category 4, using the isotropic Gaussian covarianve kernel and $k_h = 0.5$ (strong spatial correlated surfaces).

The classification results are displayed in Figure 40, from the implementation of our proposed functional statistical methodology to discriminate between strong and weak correlated Gaussian surfaces.

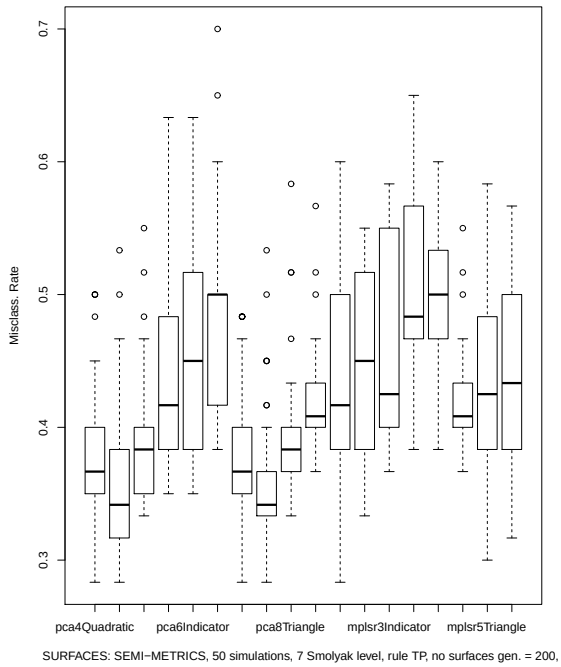

Figure 40: Results obtained with our classification methodology for random irregularities using Trapezoidal rule (level 7), using a weak spatial correlation model.

Finally, to discriminate between strong spatial correlated surfaces (smoother surfaces), the following values of parameter k_h are considered $k_h = (0.2, 0.2, 0.6, 0.6)$ (see Figures 41, 42, 43 and 44). The coresponding classification results are showed in Figure 45.

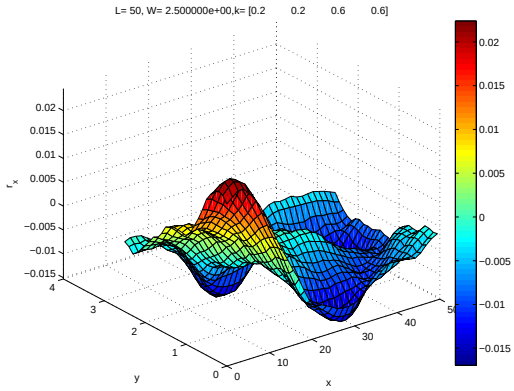

Figure 41: Random surface belongs to category 1, using the isotropic Ornstein–Uhlenbeck covariance kernel and $k_h = 0.2$ (strong spatial correlated random surface).

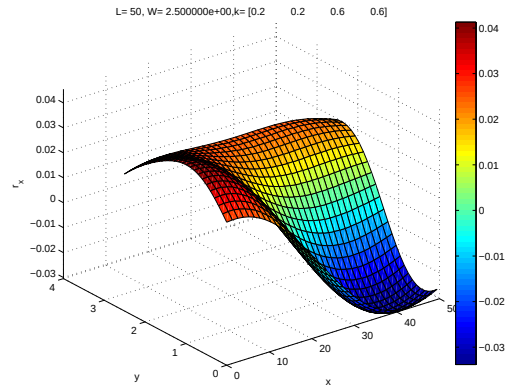

Figure 42: Random surface belongs to category 2, using the isotropic Gaussian covariance kernel and $k_h = 0.2$ (strong spatial correlated random surface).

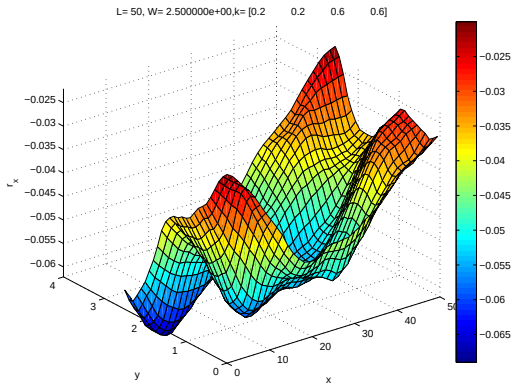

Figure 43: Random surface belongs to category 3, using the isotropic Ornstein–Uhlenbeck covariance kernel and $k_h = 0.6$ (strong spatial correlated random surface).

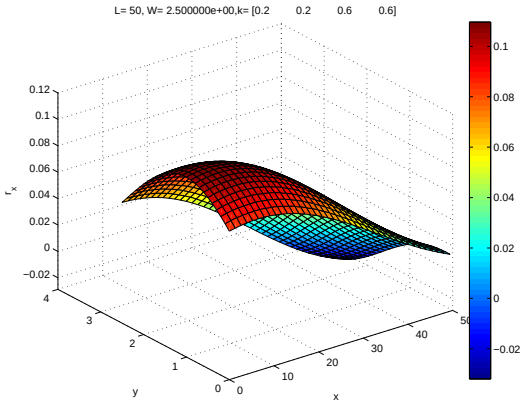

Figure 44: Random surface belongs to category 4, using the isotropic Gaussian covariance kernel and $k_h = 0.6$ (strong spatial correlated random surface).

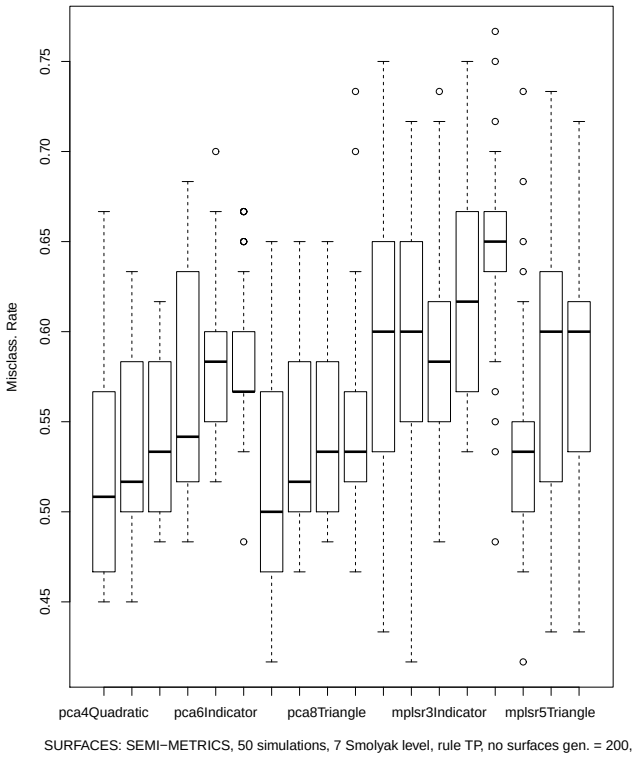

Figure 45: Results obtained with our classification methodology for random irregularities using Trapezoidal rule (level 7), from strong spatial correlated random surfaces.

We can appreciate a better performance of the functional classification methodology proposed when the non-linear Gaussian random surfaces analyzed display weak correlation, inducing a higher degree of local singularity which facilitates the detection of such a more pronounced roughness in the railway track.

VII. CONCLUSIONS

In all our implementations, different kernels have been considered, as quadratic, indicator and triangle kernels; and different inputs have been used. We improved the accuracy when we increase the number of evaluations in the Smolyak quadrature rule in both Trapezoidal and Clenshaw Curtis rule.

In Sections IV. and V., we obtain a better performance using Trapezoidal rule. This is explained by the fact that the Clenshaw-Curtis quadrature rule is a truncated expanding in the series of trigonometric functions; thus, it looks natural that we obtain less accuracy. This becames increasingly evident using FPCA semi-metric (see also Section VI.). With these results, we notice that the choice of univariate quadrature rule is not as trivial as it might seem at first sight. Such as the FPCA semi-metric only depends on the data, its accuracy is more affected by the choice of nodes. Meanwhile, the MPLSR semi-metric also depends on responses that are not affected by the quadrature rule. For that reason, FPLSR semi-metric provides us a better performance than FPCA case. Note also that that the semi-metric based on derivatives is the more accuracy (see Section IV.).

One can observe that with greater interpolation step (see Section IV.) or a finer grid (see Sections V. and VI.), a slight improvement is obtained due to associated interpolation error and weight allocation error.

In Section V., note also that the two categories distinguished in surfaces classification are very close. Also, for well-differentiated categories or groups, FPLSR outperforms FPCA when low-quality data are available or when numerical integration rules are applied at low resolution levels.

The noisy non-random irregularities studied in Section VI. provide us 12 categories to discriminate, corresponding to 12 irregularity models M_i, $i = 1, \ldots, 12$. Despite having a large number of categories and the closeness between perturbed surfaces, Figure 29 show us a good performance of our algorithm, such as we obtain a relatively low missclassification rate. In the light of the results shown in Figure 30, it was concluded that the smaller the distance A between irregularities, the greater the missclassification rate is, since they are more difficult to distinguish to each other.

The surface classification problem addressed in Section VI. leads us to the following general conclusion: The best performance of our proposed functional classification methodology is obtained when deterministic surfaces perturbed by additive Gaussian white noise are considered (non-linear model with random perturbation). While, in the Gaussian random surface case considered, a better performance is achieved when weak spatial correlated surfaces must be discriminated against strong spatial correlated surfaces. On the other hand, the worst performance is observed when we have to discriminate between smoother random surfaces, corresponding to strong spatial correlated zero-mean Gaussian surfaces.

Summarizing, the results obtained in the real-data example considered, and in the simulation study undertaken support the conclusion that our proposed functional classification al-

gorithm for n-dimensional supported non-linear random and deterministic functions offer an extended version of the previous one derived in [8], in a more flexible way, in particular, addressing the problem of random surface classification.

REFERENCES

[1] J. Aach and G.M. Church, Alignment gene expression time series with time warping algorithms, Bioinformatics 17, pp. 495–508 (2001).

[2] C. Abraham, P.A. Cornillon, E. Matzner-Løber and N. Molinari, Unsupervised curve clustering using B-splines. Scand. J. of Stat. 30, pp. 581—595 (2003).

[3] A.M. Alonso, D. Casado and J. Romo, Supervised classification for functional data: A weighted distance approach, Comp. Stat. & Data Anal. 56, pp. 2334—2346 (2012).

[4] A. Berlinet, G. Biau and L. Rouvière, Functional supervised classification with wavelets, Annales de l'ISUP 52, pp. 61—80 (2008).

[5] G. Biau, F. Bunea and M.H. Wegkamp, Functional classification in Hilbert spaces, IEEE Trans. Inf. Theor. 1, pp. 1—8 (2003).

[6] C. de Boor, A practical guide to splines, Springer, New York (1978).

[7] J. Burkardt, MATLAB Source Codes, retrieved from http://people.sc.fsu.edu/jburkardt, Florida (2011).

[8] F. Ferraty and P. Vieu, Nonparameric Functional Data Analysis, Springer, New York (2006).

[9] L. Frýba, Vibration of solids and structures under moving loads, Thomas Telford Ltd., London (1999).

[10] M. Ge and L. Fan, Learning Optimal Kernel for Pattern Classification, WSEAS Transactions on Mathematics 5, pp. 491–500 (2013).

[11] T. Gerstner and M. Griebel, Numerical Integration using Sparse Grids, Numer. Algorithms 18, pp. 209–232 (1998).

[12] T. Gerstner, Sparse Grid Quadrature Methods for Comp. Finance, University of Bonn,lecture notes (2007).

[13] G. Golub and J. Welsch, Calculation of Gauss Quadrature Rules, Mathematics of Computation 23, pp. 221–230 (1969).

[14] D. Ghosh and N. Kaabouch, A Survey on Remote Sensing Scene Classification Algorithms, WSEAS Transactions on Signal Processing 10, pp. 504–519 (2014).

[15] J.F. Hair, W.C. Black, B.J. Babin and E.R. Anderson, Multivariate data analysis, Pearson Prentice Hall, New Jersey (2006).

[16] P. Hall, D. Poskitt and B. Presnell, A functional data-analytic approach to signal discrimination, Technometrics 43, pp. 1–9 (2001).

[17] J.E. Jackson, A user's guide to Principal Components, Wiley, New York (2003).

[18] G.M. James and T.J. Hastie, Functional linear discriminant analysis for irregular sampled curves, J. Roy. Stat. Soc. Ser. B 63, pp. 533–550 (2001).

[19] G.M. James and C.A. Sugar, Clustering for sparsely sampled functional data, J. Amer. Stat. Soc. 98, pp. 397–408 (2003).

[20] V.Kaarnioja, Smolyak Quadrature, Master's Thesis, Department of Mathematics and Statistics, University of Helsinki (2013).

[21] T. Kato, Non–stationary flows of viscous and ideal fluids in $\mathbb{R}^3$, J. Func. Anal. 9, pp. 296–305 (1972).

[22] P. Kythe and M. Schaeferkotter, Handbook of Computational Methods for Integration, Chapman and Hall (2004).

[23] X. Leng and H.G. Müller, Time ordering of gene coexpression, Biostatistics 7, pp. 569–584 (2006).

[24] X.L. Liu and H.G. Müller, Modes and clustering for time-warped gene expression profile data, Bioinformatics 19, pp. 1937–1944 (2003).

[25] S. Mohammadzadeh, S.A. Mosayebi and R. Moosapoor, Investigating on the Effects of Random Irregularities of Railway Track by Half-Bogie Model, International Journal of Advances in Railway Engineering 1, pp. 61–75 (2013).

[26] H.G. Müller and U. Stadtmüller, Generalized functional linear models, Ann. Stat. 33, pp. 774–805 (2005).

[27] E.A. Nadaraya, On Estimating Regression, Theory Proba. Appl. 9, pp. 141–142 (1964).

[28] E. Novak and K. Ritter, The curse of dimension and a universal method for numerical integration, Multivariate Approx. and Splines, pp. 177–188 (1998).

[29] O.O. Oladunni, Regularized Least Squares Piecewise Multi-classification Machine, WSEAS Transactions on Computers 12, pp. 18–27 (2013).

[30] T. Patterson, The Optimal Addition of Points to Quadrature Formulae, Mathematics of Computation 22, pp. 847–856 (1968).

[31] J.O. Ramsay and B.W. Silverman, Functional Data Analysis, Springer–Verlag, New York (2005).

[32] M. Rincón and M.D. Ruiz-Medina, Local wavelet-vaguelette-based functional classification of gene expression data, Biometrical J. 54, pp. 75–93 (2012).

[33] M.D. Ruiz-Medina, Spatial autoregressive and moving average Hilbertian processes, J. Multiv. Anal. 102, pp. 292–305 (2011).

[34] M.D. Ruiz-Medina, Spatial functional prediction from spatial autoregressive Hilbertian processes, Environmetrics 23, pp. 119–128 (2012).

[35] M.D. Ruiz-Medina and R.M. Espejo, Spatial autoregressive functional plug-in prediction of ocean surface temperature, Stoch. Env. Res. Risk A. 26, pp. 335–344 (2012).

[36] M.D. Ruiz-Medina, V.V. Anh, R.M. Espejo and M.P. Frías M.P., Heterogeneous spatial dynamical regression in a Hilbert-valued context, Stoch. Anal. Appl. 31, pp. 509–527 (2013).

[37] P. Schneider, NURB Curves: A Guide for the Uninitiated, MACTECH (2014).

[38] I.J. Schoenberg,Spline Functions and the Problem of Graduation, Proc. of the Nat. Acad. of Sciences of the United States of America 52, pp. 947—950 (2012).

[39] B. Schölkopf and A. Smola, Learning with Kernels, MIT Press, Cambridge (2002).

[40] M. Schumaker, Splines functions: basic theory, Wiley, New York (1981).

[41] A. Stroud and D. Secrest, Gaussian Quadrature Formulas, Prentice Hall, New Jersey (1966).

[42] G. Wasilkowski and H. Wozniakowski, Explicit Cost Bounds of Algorithms for Multivariate Tensor Product Problems, J. of Complexity 11, pp. 1–56 (1995).

[43] G.S. Watson, Smooth regression analysis, Sankhya Ser. A 26, pp. 359–372 (1964).

[44] Y.B. Yang, J.D. Yau and Y.S. Wu, Vehicle-Bridge interaction dynamics with applications to high-speed railways, World scientific publishing Co Pte Ltd., Singapore (2004).

[45] J.Y. Yang, Z.L. Peng, Z. Yu, R.J. Zhang, V. Anh and D. Wang D., Prediction of protein structural classes by recurrence quantification analysis based on chaos game representation, J. Theor. Biol. 257, pp. 618–626 (2009).

[46] J.Y. Yang, Z. Yu and V. Anh, Clustering structures of large proteins using multifractal analyses based on a 6-letter model and hydrophobicity scale of amino acids, Chaos, Solitons and Fractals 40, pp. 607–620 (2009).

[47] K. Youcef, T. Sabiha, D. El Mostafa, D. Ali and M. Bachir, Dynamic analysis of train-bridge system and riding comfort of trains with rail irregularities, Journal of Mechanical Science and Technology 27, pp. 951–962 (2012).