STATISTICS IN TRANSITION new series, Spring 2015
Vol. 16, No. 1, pp. 97–110

CLASSIFICATION PROBLEMS BASED ON REGRESSION MODELS FOR MULTI-DIMENSIONAL FUNCTIONAL DATA

Tomasz Górecki[1], **Mirosław Krzyśko**[2], **Waldemar Wołyński**[3]

ABSTRACT

Data in the form of a continuous vector function on a given interval are referred to as multivariate functional data. These data are treated as realizations of multivariate random processes. We use multivariate functional regression techniques for the classification of multivariate functional data. The approaches discussed are illustrated with an application to two real data sets.

Key words: multivariate functional data, functional data analysis, multivariate functional regression, classification.

1. Introduction

Much attention has been paid in recent years to methods for representing data as functions or curves. Such data are known in the literature as functional data (Ramsay and Silverman (2005)). Applications of functional data can be found in various fields, including medicine, economics, meteorology and many others. In many applications there is a need to use statistical methods for objects characterized by multiple features observed at many time points (doubly multivariate data). Such data are called multivariate functional data. The pioneering theoretical work was that of Besse (1979), in which random variables take values in a general Hilbert space. Saporta (1981) presents an analysis of multivariate functional data from the point of view of factorial methods (principal components and canonical analysis). In this paper we focus on the problem of classification via regression for multivariate functional data. Functional regression models have been extensively studied; see for example James (2002), Müller and

[1] Faculty of Mathematics and Computer Science, Adam Mickiewicz University, Poland. E-mail: tomasz.gorecki@amu.edu.pl.
[2] Faculty of Mathematics and Computer Science, Adam Mickiewicz University, Poland. E-mail: mkrzysko@amu.edu.pl.
[3] Faculty of Mathematics and Computer Science, Adam Mickiewicz University, Poland. E-mail: wolynski@amu.edu.pl.

Stadmüller(2005), Reiss and Ogden (2007), Matsui et al. (2008) and Li et al. (2010). Various basic classification methods have also been adapted to functional data, such as linear discriminant analysis (Hastie et al. (1995)), logistic regression (Rossi et al. (2002)), penalized optimal scoring (Ando (2009)), $k$nn (Ferraty and Vieu (2003)), SVM (Rossi and Villa (2006)), and neural networks (Rossi et al. (2005)). Moreover, the combining of classifiers has been extended to functional data (Ferraty and Vieu (2009)).

In the present work we adapt multivariate regression models to the classification of multivariate functional data. We focus on the binary classification problem. There exist several techniques for extending the binary problem to multi-class classification problems. A brief overview can be found in Krzyśko and Wołyński (2009). The accuracy of the proposed methods is demonstrated using biometrical examples. Promising results were obtained for future research.

2. Classification problem

The classical classification problem involves determining a procedure by which a given object can be assigned to one of K populations based on observation of p features of that object.

The object being classified can be described by a random pair $(\boldsymbol{X}, Y)$, where $\boldsymbol{X} = (X_1, X_2, \ldots, X_p)' \in \mathbf{R}^p$ and $Y \in \{0, 1, \ldots, K-1\}$.

The optimum Bayesian classifier then takes the form (Anderson (1984)):

$$d(\boldsymbol{x}) = \arg \max_{k \in \{0,1,\ldots,K-1\}} \mathrm{P}(Y = k | \boldsymbol{X} = \boldsymbol{x}).$$

We shall further consider only the case $K = 2$. Here

$$d(\boldsymbol{x}) = \begin{cases} 1, & \mathrm{P}(Y = 1|\boldsymbol{X} = \boldsymbol{x}) \geqslant \mathrm{P}(Y = 0|\boldsymbol{X} = \boldsymbol{x}); \\ 0, & \mathrm{P}(Y = 1|\boldsymbol{X} = \boldsymbol{x}) < \mathrm{P}(Y = 0|\boldsymbol{X} = \boldsymbol{x}). \end{cases}$$

We note that

$$\mathrm{P}(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = \mathrm{E}(Y|\boldsymbol{X} = \boldsymbol{x}) = r(\boldsymbol{x}),$$

where $r(\boldsymbol{x})$ is the regression function of the random variable Y with respect to the random vector $\boldsymbol{X}$.

Hence

$$d(\boldsymbol{x}) = \begin{cases} 1, & r(\boldsymbol{x}) \geqslant 1/2; \\ 0, & r(\boldsymbol{x}) < 1/2. \end{cases}$$

3. Functional data

We now assume that the object being classified is described by a p-dimensional random process $\boldsymbol{X} = (X_1, X_2, ..., X_p)' \in L_2^p(I)$, where $L_2(I)$ is the Hilbert space of square-integrable functions.

Let $\boldsymbol{x}$be the realization of the random process $\boldsymbol{X}$. Moreover, assume that the k th component of the vector $\boldsymbol{x}$ can be represented by a finite number of orthonormal basis functions $\{\varphi_b\}$

$$x_k(t) = \sum_{b=0}^{B_k} c_{kb}\varphi_b(t),\ t \in I,\ k = 1, \ldots, p, \tag{1}$$

where $c_{k0}, c_{k1}, \ldots, c_{kB_k}$ are the unknown coefficients.

Let $\boldsymbol{c} = (c_{10}, \ldots, c_{1B_1}, \ldots, c_{p0}, \ldots, c_{pB_p})'$
and

$$\boldsymbol{\Phi}(t) = \begin{bmatrix} \boldsymbol{\varphi}_1'(t) & \boldsymbol{0} & \ldots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\varphi}_2'(t) & \ldots & \boldsymbol{0} \\ \ldots & \ldots & \ldots & \ldots \\ \boldsymbol{0} & \boldsymbol{0} & \ldots & \boldsymbol{\varphi}_p'(t) \end{bmatrix},$$

where $\boldsymbol{\varphi}_k(t) = (\varphi_0(t), ..., \varphi_{B_k}(t))', k = 1, ..., p$.

Then, the vector of the continuous function $\boldsymbol{x}$ at point $t$can be represented as

$$\boldsymbol{x}(t) = \boldsymbol{\Phi}(t)\boldsymbol{c}. \tag{2}$$

We can estimate the vector $\boldsymbol{c}$ on the basis of n independent realizations $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ of the random process $\boldsymbol{X}$ (functional data).

Typically data are recorded at discrete moments in time. Let x_{kj} denote an observed value of the feature X_k, $k = 1, 2, \ldots p$ at the jth time point t_j, where $j = 1, 2, ..., J$. Then our data consist of the pJ pairs (t_j, x_{kj}). These discrete data can be smoothed by continuous functions x_k and I is a compact set such that $t_j \in I$, for $j = 1, ..., J$.

Details of the process of transformation of discrete data to functional data can be found in Ramsay and Silverman (2005) or in Górecki et al. (2014).

4. Regression analysis for functional data

We now consider the problem of the estimation of the regression function $r(\boldsymbol{x})$.

Let us assume that we have an n-element training sample

$$\boldsymbol{x}(t) = \boldsymbol{\Phi}(t)\boldsymbol{c}. \tag{3}$$

where $\boldsymbol{x}_i \in L_2^p(I)$ and $y_i \in \{0,1\}$.

Analogously as in section 3, we assume that the functions $\boldsymbol{x}_i$ are obtained as the result of a process of smoothing n independent discrete data pairs (t_j, x_{kij}), $k = 1,\ldots,p$, $j = 1,...,J$, $i = 1,...,n$.

Thus the functions $\boldsymbol{x}_i$ at point t have the following representation:

$$\boldsymbol{x}_i(t) = \boldsymbol{\Phi}(t)\boldsymbol{c}_i,\ i = 1,2,\ldots,n. \tag{4}$$

4.1. Multivariate linear regression. We take the following model for the regression function:

$$r(\boldsymbol{x}) = \beta_0 + <\boldsymbol{\beta},\boldsymbol{x}> = \beta_0 + \int_I \boldsymbol{\beta}'(t)\boldsymbol{x}(t)dt.$$

We seek the unknown parameters in the regression function by minimizing the sum of squares

$$S(\beta_0,\boldsymbol{\beta}) = \sum_{i=1}^{n}(y_i - \beta_0 - \int_I \boldsymbol{\beta}'(t)\boldsymbol{x}_i(t)dt)^2.$$

We assume that the functions $\boldsymbol{x}_i$, $i = 1,2,\ldots,n$ have the representation (4). We adopt an analogous representation for the p-dimensional weighting function $\boldsymbol{\beta}$, namely

$$\boldsymbol{\beta}(t) = \boldsymbol{\Phi}(t)\boldsymbol{d}, \tag{5}$$

where $\boldsymbol{\beta} = (\beta_1,\beta_2,\ldots,\beta_p)'$,and $\boldsymbol{d} = (d_{10},\ldots,d_{1B_1},\ldots,d_{p0},\ldots,d_{pB_p})'$.

Then

$$\begin{aligned}\int_I \boldsymbol{\beta}'(t)\boldsymbol{x}_i(t)dt &= \int_I \boldsymbol{d}'\boldsymbol{\Phi}'(t)\boldsymbol{\Phi}(t)\boldsymbol{c}_i dt \\ &= \boldsymbol{d}'\int_I \boldsymbol{\Phi}'(t)\boldsymbol{\Phi}(t)dt\boldsymbol{c}_i = \boldsymbol{d}'\boldsymbol{c}_i, \quad i = 1,2,\ldots,n.\end{aligned}$$

Hence

$$S(\beta_0,\boldsymbol{\beta}) = S(\beta_0,\boldsymbol{d}) = \sum_{i=1}^{n}(y_i - \beta_0 - \boldsymbol{d}'\boldsymbol{c}_i)^2.$$

We define $\boldsymbol{y} = (y_1,y_2,\ldots,y_n)'$ and

$$\boldsymbol{Z} = \begin{bmatrix} 1 & \boldsymbol{c}_1' \\ 1 & \boldsymbol{c}_2' \\ \vdots & \vdots \\ 1 & \boldsymbol{c}_n' \end{bmatrix}, \quad \boldsymbol{\gamma} = \begin{bmatrix} \beta_0 \\ \boldsymbol{d} \end{bmatrix}.$$

Then

$$S(\beta_0, \boldsymbol{\beta}) = S(\boldsymbol{\gamma}) = (\boldsymbol{y} - \boldsymbol{Z\gamma})' (\boldsymbol{y} - \boldsymbol{Z\gamma}).$$

Minimizing the above sum of squares leads to the choice of a vector $\boldsymbol{\gamma}$ satisfying

$$\boldsymbol{Z}'\boldsymbol{Z}\boldsymbol{\gamma} = \boldsymbol{Z}'\boldsymbol{y}. \tag{6}$$

Provided the matrix $\boldsymbol{Z}'\boldsymbol{Z}$ is non-singular, equation (6) has the unique solution

$$\hat{\boldsymbol{\gamma}} = (\boldsymbol{Z}'\boldsymbol{Z})^{-1}\boldsymbol{Z}'\boldsymbol{y}. \tag{7}$$

In the case of functional data we may use the smoothed least squares method (Ramsay and Silverman (2005)), that is, we minimize the sum of squares in the form

$$S(\beta_0, \boldsymbol{\beta}) = \sum_{i=1}^{n}(y_i - \beta_0 - \int_I \boldsymbol{\beta}'(t)\boldsymbol{x}_i(t)dt)^2 + \lambda \int_I [L\boldsymbol{\beta}(t)]'[L\boldsymbol{\beta}(t)]dt,$$

where L denotes the linear differential operator. Assuming $L\boldsymbol{\beta} = D^2\boldsymbol{\beta}$, we obtain

$$L\boldsymbol{\beta}(t) = D^2(\boldsymbol{\Phi}(t)\boldsymbol{d}) = D^2(\boldsymbol{\Phi}(t))\boldsymbol{d}.$$

Thus

$$\int_I [L\boldsymbol{\beta}(t)]'[L\boldsymbol{\beta}(t)]dt = \boldsymbol{d}' \int_I [D^2\boldsymbol{\Phi}(t)]'[D^2\boldsymbol{\Phi}(t)]dt\, \boldsymbol{d}.$$

We define

$$\boldsymbol{R} = \int_I [D^2\boldsymbol{\Phi}(t)]'[D^2\boldsymbol{\Phi}(t)]dt.$$

Hence

$$\int_I [L\boldsymbol{\beta}(t)]'[L\boldsymbol{\beta}(t)]dt = \boldsymbol{d}'\boldsymbol{R}\boldsymbol{d} = \boldsymbol{\gamma}'\boldsymbol{R}_0\gamma,$$

where

$$\boldsymbol{R}_0 = \begin{bmatrix} 0 & \boldsymbol{0}' \\ \boldsymbol{0} & \boldsymbol{R} \end{bmatrix}.$$

Then

$$S(\beta_0, \boldsymbol{\beta}) = S(\boldsymbol{\gamma}) = (\boldsymbol{y} - \boldsymbol{Z\gamma})' (\boldsymbol{y} - \boldsymbol{Z\gamma}) + \lambda\boldsymbol{\gamma}'\boldsymbol{R}_0\gamma.$$

Minimizing the above sum of squares leads to the choice of a vector $\boldsymbol{\gamma}$ satisfying the equation

$$(\boldsymbol{Z}'\boldsymbol{Z} + \lambda\boldsymbol{R}_0)\boldsymbol{\gamma} = \boldsymbol{Z}'\boldsymbol{y}.$$

The equation thus obtained has the form

$$\hat{\boldsymbol{\gamma}} = (\boldsymbol{Z}'\boldsymbol{Z} + \lambda\boldsymbol{R}_0)^{-1}\boldsymbol{Z}'\boldsymbol{y}. \tag{8}$$

From this we obtain the following form for the estimator of the regression function for the multivariate functional data:

$$\hat{r}(\boldsymbol{x}) = \hat{\beta}_0 + \hat{\boldsymbol{d}}'\boldsymbol{c},$$

where $\hat{\boldsymbol{\gamma}} = (\hat{\beta}_0, \hat{\boldsymbol{d}})'$ is given by the formula (7) or (8).

4.2. Functional logistic regression. We adopt the following logistic regression model for functional data:

$$r(\boldsymbol{x}) = \frac{\exp(\beta_0 + < \boldsymbol{\beta}, \boldsymbol{x} >)}{1 + \exp(\beta_0 + < \boldsymbol{\beta}, \boldsymbol{x} >)} = \frac{\exp(\beta_0 + \int_I \boldsymbol{\beta}'(t)\boldsymbol{x}(t)dt)}{1 + \exp(\beta_0 + \int_I \boldsymbol{\beta}'(t)\boldsymbol{x}(t)dt)}. \tag{9}$$

Using the representation of the function $\boldsymbol{x}$ given by (2) and the weighting function $\boldsymbol{\beta}$ given by (5) we reduce (9) to a standard logistic regression model in the form

$$r(\boldsymbol{x}) = \frac{exp(\beta_0 + \boldsymbol{d}'\boldsymbol{c})}{1 + exp(\beta_0 + \boldsymbol{d}'\boldsymbol{c})}.$$

To estimate the unknown parameters of the model, we use the training sample $\mathcal{L}_n$ and the analogous representation for the functions $\boldsymbol{x}_i$, $i = 1, 2, \ldots, n$ given by (4).

Thus we obtain the following form for the estimator of the regression function

$$\hat{r}(\boldsymbol{x}) = \frac{\exp(\hat{\beta}_0 + \hat{\boldsymbol{d}}'\boldsymbol{c})}{1 + \exp(\hat{\beta}_0 + \hat{\boldsymbol{d}}'\boldsymbol{c})}.$$

4.3. Local linear regression smoothers. We consider the problem of nonparametric estimation of a regression function $r(\boldsymbol{x})$ from a sample (3).

Let $\boldsymbol{x}_0$ be a fixed and known point in the space $L_2^p(I)$.

Using Taylor series, we can approximate $r(\boldsymbol{x})$, where $\boldsymbol{x}$ is close to a point $\boldsymbol{x}_0$, as follows:

$$r(\boldsymbol{x}) \approx r(\boldsymbol{x}_0) + < \frac{\partial r(\boldsymbol{x}_0)}{\partial \boldsymbol{x}_0}, \boldsymbol{x} - \boldsymbol{x}_0 > = \beta_0 + < \boldsymbol{\beta}, \boldsymbol{x} - \boldsymbol{x}_0 >, \tag{10}$$

where

$$\beta_0 = r(\boldsymbol{x}_0), \quad \boldsymbol{\beta} = \frac{\partial r(\boldsymbol{x}_0)}{\partial \boldsymbol{x}_0}.$$

This is a local polynomial regression problem in which we use the data to estimate the polynomial which best approximates $r(\boldsymbol{x})$ in a small neighborhood around the point $\boldsymbol{x}_0$, i.e. we minimize it with respect to β_0 and $\boldsymbol{\beta}$ in the function

$$S(\beta_0, \boldsymbol{\beta}) = \sum_{i=1}^{n} \left(y_i - \beta_0 - \int_I \boldsymbol{\beta}'(t)(\boldsymbol{x}_i(t) - \boldsymbol{x}_0(t))dt \right)^2 K(\frac{1}{h}||\boldsymbol{x}_i - \boldsymbol{x}_0||).$$

This is a weighted least squares problem where the weights are given by the kernel functions $K(||\boldsymbol{x}_i - \boldsymbol{x}_0||/h)$.

Analogously as in the previous sections, suppose that the vector functions $\boldsymbol{x}_i$ and $\boldsymbol{\beta}$ are in the same space, i.e.

$$\boldsymbol{x}_i(t) = \boldsymbol{\Phi}(t)\boldsymbol{c}_i, \ i = 1, 2, \ldots, n, \quad \boldsymbol{\beta}(t) = \boldsymbol{\Phi}(t)\boldsymbol{d}.$$

Then

$$\int_I \boldsymbol{\beta}'(t)(\boldsymbol{x}_i(t) - \boldsymbol{x}_0(t))dt = \boldsymbol{d}'(\int_I \boldsymbol{\Phi}'(t)\boldsymbol{\Phi}(t)dt)(\boldsymbol{c}_i - \boldsymbol{c}_0) = \boldsymbol{d}'(\boldsymbol{c}_i - \boldsymbol{c}_0),$$

$$||\boldsymbol{x}_i - \boldsymbol{x}_0|| = \sqrt{(\boldsymbol{c}_i - \boldsymbol{c}_0)'(\boldsymbol{c}_i - \boldsymbol{c}_0)}, \ i = 1, 2, \ldots, n.$$

The least squares problem is then to minimize the weighted sum-of-squares function

$$S(\beta_0, \boldsymbol{\beta}) = S(\beta_0, \boldsymbol{d}) = \sum_{i=1}^{n} (y_i - \beta_0 - \boldsymbol{d}'(\boldsymbol{c}_i - \boldsymbol{c}_0))^2 K(\frac{1}{h}\sqrt{(\boldsymbol{c}_i - \boldsymbol{c}_0)'(\boldsymbol{c}_i - \boldsymbol{c}_0)})$$

with respect to the parameters β_0 and $\boldsymbol{d}$.

It is convenient to define the following vectors and matrices:

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \boldsymbol{Z} = \begin{bmatrix} 1 & (\boldsymbol{c}_1 - \boldsymbol{c}_0)' \\ 1 & (\boldsymbol{c}_2 - \boldsymbol{c}_0)' \\ \vdots & \vdots \\ 1 & (\boldsymbol{c}_n - \boldsymbol{c}_0)' \end{bmatrix},$$

$$\gamma = \begin{bmatrix} \beta_0 \\ \boldsymbol{d} \end{bmatrix}, \quad \boldsymbol{W} = \begin{bmatrix} K_1 & 0 & \ldots & 0 \\ 0 & K_2 & \ldots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \ldots & K_n \end{bmatrix},$$

where

$$K_i = K(\frac{1}{h}\sqrt{(\boldsymbol{c}_i - \boldsymbol{c}_0)'(\boldsymbol{c}_i - \boldsymbol{c}_0)}),\ i = 1, 2, \dots, n.$$

The least squares problem is then to minimize the function

$$S(\beta_0, \boldsymbol{\beta}) = S(\boldsymbol{\gamma}) = (\boldsymbol{y} - \boldsymbol{Z\gamma})' \boldsymbol{W} (\boldsymbol{y} - \boldsymbol{Z\gamma}).$$

The solution is

$$\hat{\gamma} = (\boldsymbol{Z}'\boldsymbol{W}\boldsymbol{Z})^{-1}\boldsymbol{Z}'\boldsymbol{W}\boldsymbol{y}$$

provided $\boldsymbol{Z}'\boldsymbol{W}\boldsymbol{Z}$ is a non-singular matrix.

As in the case of multivariate functional linear regression model we can also include an additional smoothing component. Then, we seek the unknown parameter $\boldsymbol{\gamma}$ by minimizing the sum of squares

$$S(\gamma) = (\boldsymbol{y} - \boldsymbol{Z}\gamma)' \boldsymbol{W} (\boldsymbol{y} - \boldsymbol{Z}\gamma) + \lambda\gamma'\boldsymbol{R}_0\gamma.$$

Provided the matrix $\boldsymbol{Z}'\boldsymbol{W}\boldsymbol{Z} + \lambda\boldsymbol{R}_0$ is non-singular we have the unique solution

$$\hat{\gamma} = (\boldsymbol{Z}'\boldsymbol{W}\boldsymbol{Z} + \lambda\boldsymbol{R}_0)^{-1}\boldsymbol{Z}'\boldsymbol{W}\boldsymbol{y}.$$

The $r(\boldsymbol{x}_0)$ is than estimated by the fitted intercept parameter (i.e. by $\hat{\boldsymbol{\beta}}_0$) as this defines the position of the estimated local polynomial curve at the point $\boldsymbol{x}_0$. By varying the value of $\boldsymbol{x}_0$, we can build up an estimate of the function $r(\boldsymbol{x})$ over the range of the data.

We have

$$\hat{r}(\boldsymbol{x}_0) = \boldsymbol{e}'\hat{\boldsymbol{\gamma}},$$

where the vector $\boldsymbol{e}$ is of the length $B_1 + \cdots + B_p + p + 1$ and has a 1 in the first position and 0's elsewhere.

4.4. Nadaraya-Watson kernel estimator. In Section 4.3 we approximated the regression function $r(\boldsymbol{x})$ using Taylor series. In the approximation (10) let us take into account only the first term, i.e.

$$r(\boldsymbol{x}) \approx r(\boldsymbol{x}_0) = \beta_0.$$

Then

$$S(\beta_0) = \sum_{i=1}^{n}(y_i - \beta_0)^2 K(\frac{1}{h}||\boldsymbol{x}_i - \boldsymbol{x}_0||).$$

Minimizing the above sum of squares leads to the kernel estimator of the regression function $r(\boldsymbol{x})$ of the form

$$\hat{r}(\boldsymbol{x}) = \frac{\sum_{i=1}^{n} y_i K_i}{\sum_{i=1}^{n} K_i},$$

where $K_i = K\left(\frac{1}{h}\sqrt{(\boldsymbol{c}_i - \boldsymbol{c}_0)'(\boldsymbol{c}_i - \boldsymbol{c}_0)}\right),\ i = 1, 2, \ldots, n.$

This gives us a well-known kernel estimator proposed by Nadaraya and Watson (1964).

5. Examples

Experiments were carried out on two data sets, these being labelled data sets whose labels are given. The data sets originate from Olszewski (2001).The *ECG* data set uses two electrodes (Figure 1) to collect data during one heartbeat. Each heartbeat is described by a multivariate time series (MTS) sample with two variables and an assigned classification of normal or abnormal. Abnormal heartbeats are representative of a cardiac pathology known as supraventricular premature beat. The *ECG* data set contains 200 MTS samples, of which 133 are normal and 67 are abnormal. The length of an MTS sample is between 39 and 152.

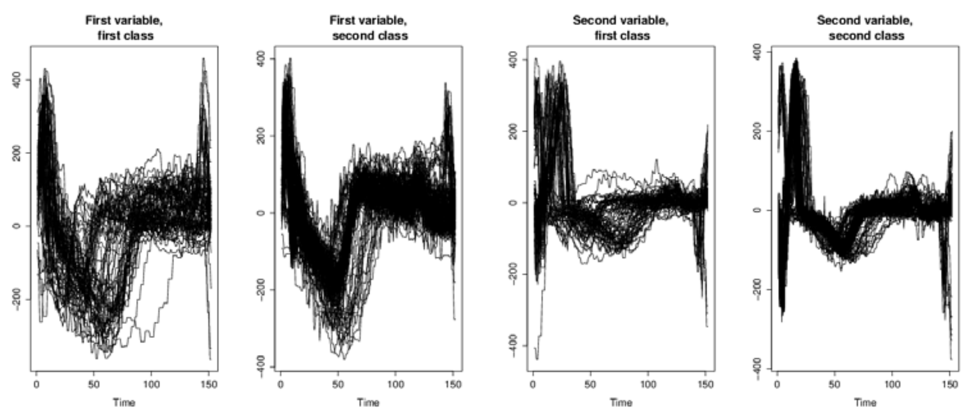

Figure 1. Variables of the extended *ECG* data set.

The *Wafer* data set uses six vacuum-chamber sensors (Figure 2) to collect data while monitoring an operational semiconductor fabrication plant. Each wafer is described by an MTS sample with six variables and an assigned classification of normal or abnormal. The data set used here contains 327 MTS samples, of which 200 are normal and 127 are abnormal. The length of an MTS sample is between 104 and 198.

The multivariate samples in the data sets are of different lengths. For each data set, the multivariate samples are extended to the length of the longest multivariate sample in the set (Rodriguez et al. (2005)). We extend all variables to the same length. For a short univariateinstance x with length J, we extend it to a long instance x_{ex} with length $J_{\max}$ by setting

$$x_{ex}(t_j) = x(t_i), \quad \text{for} \quad i = \left\lceil \frac{j-1}{J_{\max}-1}(J-1)+0.5 \right\rceil \quad (j = 1, 2, \ldots, J_{\max}).$$

Some of the values in a data sample are duplicated in order to extend the sample. For instance, if we wanted to extend a data sample of length 75 to a length of 100, one out of every three values would be duplicated. In this way, all of the values in the original data sample are contained in the extended data sample.

For the classification process, we used the classifiers described above. For each data set we calculated the classification error rate using the leave-one-out cross-validation method (LOO CV). Table 1 contains the results of the classification error rates (in %).

Table 1. Classification error (in %)

Model	**ECG**	**Wafer**
Multivariate functional linear regression	11.50	0.59
Functional logistic regression	11.50	0.17
Local linear regression smoothers	16.50	0.67
Nadaraya-Watson kernel estimator	20.50	10.64

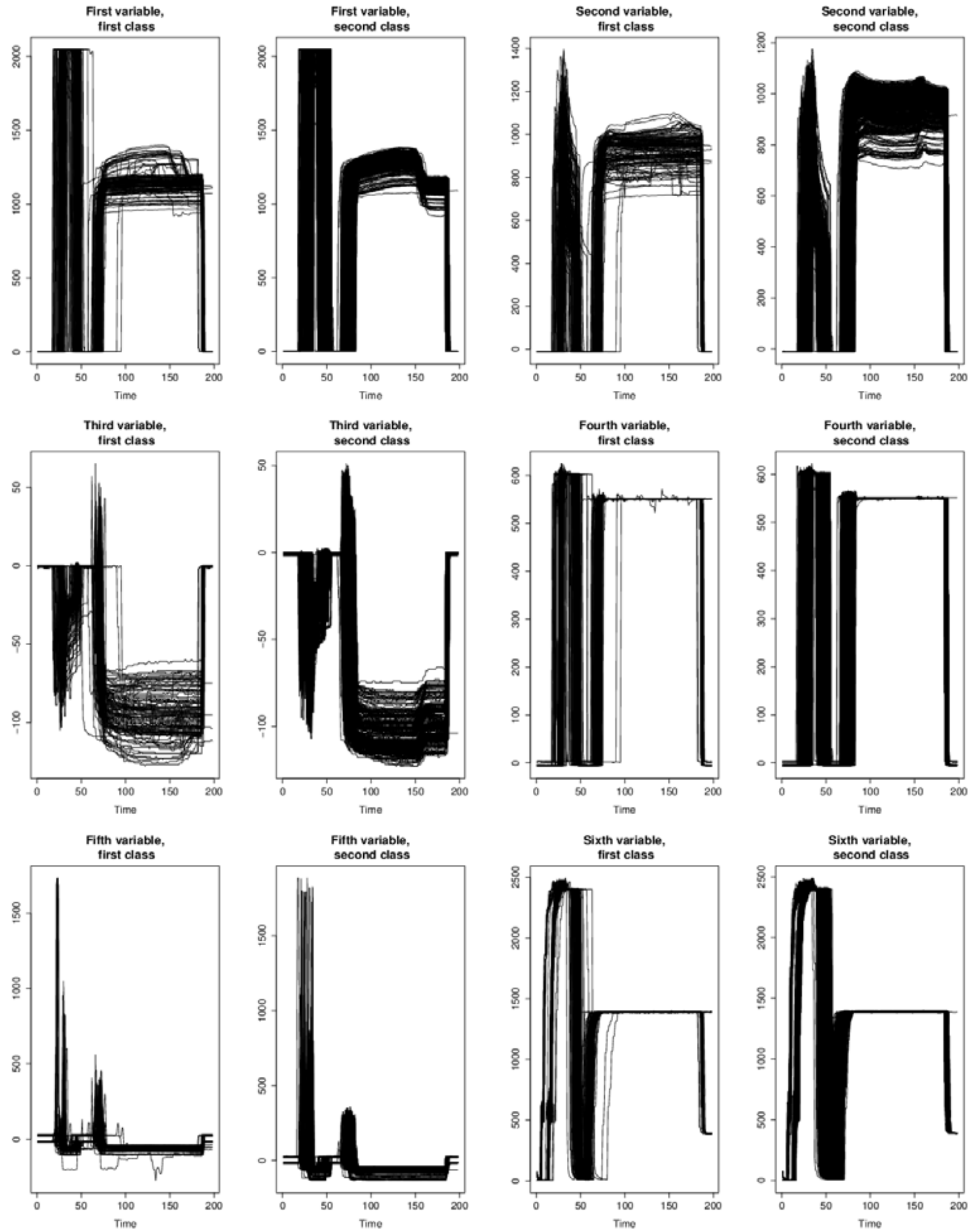

Figure 2. Variables of the extended *Wafer* data set.

From Table 1 we see that the *ECG* data set is difficult to recognize. None of the four regression methods can deal with it well. In contrast, the data set *Wafer* is easily recognizable. For this set of data definitely the best results are given by a functional logistic regression. We also see a big difference between the local linear regression smother, and the Nadaraya-Watson kernel estimator.

6. Conclusion

This paper develops and analyzes methods for constructing and using regression methods of classification for multivariate functional data. These methods were applied to two biometrical multivariate time series. In the case of these examples it was shown that the use of multivariate functional regression methods for classification gives good results. Of course, the performance of the algorithm needs to be further evaluated on additional real and artificial data sets. In a similar way, we can extend other regression methods, such as partial least squares regression – PLS (Wold (1985)), least absolute shrinkage and selection operator – LASSO (Tibshirani (1996)), or least-angle regression – LARS (Efron et al. (2004)), to the multivariate functional case. This will be the direction of our future research.

REFERENCES

ANDERSON, T. W., (1984). An Introduction to Multivariate Statistical Analysis. Wiley, New York.

ANDO, T., (2009). Penalized optimal scoring for the classification of multi-dimensional functional data. Statistcal Methodology 6, 565–576.

BESSE, P., (1979). Etude descriptive d'un processus. Ph.D. thesis, Université Paul Sabatier.

EFRON, B., HASTIE, T., JOHNSTONE, I., TIBSHIRANI, R., (2004). Least Angle Regression. Annals of Statistics 32(2), 407–499.

FERRATY, F., VIEU, P., (2003). Curve discrimination. A nonparametric functional approach. Computational Statistics & Data Analysis 44, 161–173.

FERRATY, F., VIEU, P., (2006). Nonparametric Functional Data Analysis: Theory and Practice. Springer, New York.

FERRATY, F., VIEU, P., (2009). Additive prediction and boosting for functional data. Computational Statistics & Data Analysis 53(4), 1400–1413.

GÓRECKI, T., KRZYŚKO, M., (2012). Functional Principal Components Analysis. In: J. Pociecha and R. Decker (Eds.): Data analysis methods and its applications. C. H. Beck, Warszawa, 71–87.

GÓRECKI, T, KRZYŚKO, M., WASZAK, Ł., WOŁYŃSKI, W., (2014). Methods of reducing dimension for functional data. Statistics in Transition new series 15, 231–242.

HASTIE, T. J., TIBSHIRANI, R. J., BUJA, A., (1995). Penalized discriminant analysis. Annals of Statistics 23, 73–102.

JAMES, G. M., (2002). Generalized linear models with functional predictors. Journal of the Royal Statistical Society 64(3), 411–432.

JACQUES, J., PREDA, C., (2014). Model-based clustering for multivariate functional data. Computational Statistics & Data Analysis 71, 92–106.

KRZYŚKO, M., WOŁYŃSKI, W., (2009). New variants of pairwise classification. European Journal of Operational Research 199(2), 512–519.

MATSUI, H., ARAKI, Y., KONISHI, S., (2008). Multivariate regression modeling for functional data. Journal of Data Science 6, 313–331.

MÜLLER, H. G., STADMÜLLER, U., (2005). Generalized functional linear models. Annals of Statistics 33, 774–805.

NADARAYA, E. A., (1964). On Estimating Regression. Theory of Probability and its Applications 9(1), 141–142.

OLSZEWSKI, R. T., (2001). Generalized Feature Extraction for Structural Pattern Recognition in Time-Series Data. Ph.D. Thesis, Carnegie Mellon University, Pittsburgh, PA.

RAMSAY, J. O., SILVERMAN, B. W., (2005). Functional Data Analysis. Springer, New York.

REISS, P. T., OGDEN R. T., (2007). Functional principal component regression and functional partial least squares. Journal of the American Statistcal Assosiation 102(479), 984–996.

ROSSI, F., DELANNAYC, N., CONAN-GUEZA, B., VERLEYSENC, M., (2005). Representation of functional data in neural networks. Neurocomputing 64, 183–210.

ROSSI, F., VILLA, N., (2006). Support vector machines for functional data classification. Neural Computing 69, 730–742.

ROSSI, N., WANG, X., RAMSAY, J. O., (2002). Nonparametric item response function estimates with EM algorithm. Journal of Educational and Behavioral Statistics 27, 291–317.

RODRIGUEZ, J. J., ALONSO, C. J., MAESTRO, J. A., (2005). Support vector machines of intervalbased features for time series classification. Knowledge-Based Systems 18, 171–178.

SAPORTA, G., (1981). Méthodes exploratoires d'analyse de données temporelles, thèse de doctorat d'état es sciences mathématiques soutenue le 10 juin 1981, Université Pierre et Marie Curie.

SHMUELI, G., (2010). To explain or to predict? Statistical Science 25(3), 289–310.

TIBSHIRANI, R., (1996). Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society, Series B 58(1), 267–288.

WATSON, G. S., (1964). Smooth regression analysis. Sankhya – The Indian Journal of Statistics, Series A 26(4), 359–372.

WOLD, H., (1985). Partial least squares. In: S. Kotz, and N.L. Johnson (Eds.): Encyclopedia of statistical sciences vol. 6, Wiley, New York, 581–591.