Applied Intelligence (2021) 51:161–184
https://doi.org/10.1007/s10489-020-01765-6

Automatic feature scaling and selection for support vector machine classification with functional data

Asunción Jiménez-Cordero[1] · Sebastián Maldonado[2,3]

Published online: 6 August 2020

Abstract

FunctionalData Analysis (FDA) has become a very important field in recent years due to its wide range of applications. However, there are several real-life applications in which hybrid functional data appear, i.e., data with functional and static covariates. The classification of such hybrid functional data is a challenging problem that can be handled with the Support Vector Machine (SVM). Moreover, the selection of the most informative features may yield to drastic improvements in the classification rates. In this paper, an embedded feature selection approach for SVM classification is proposed, in which the isotropic Gaussian kernel is modified by associating a bandwidth to each feature. The bandwidths are jointly optimized with the SVM parameters, yielding an alternating optimization approach. The effectiveness of our methodology was tested on benchmark data sets. Indeed, the proposed method achieved the best average performance when compared to 17 other feature selection and SVM classification approaches. A comprehensive sensitivity analysis of the parameters related to our proposal was also included, confirming its robustness.

Keywords Feature selection · Functional data · Support vector machines · Classification · Feature scaling

1 Introduction

Functional Data Analysis (FDA)has become an outstanding field in recent years [36, 71, 72]. Instead of assuming scalar covariates, FDA handles problems in which the data samples correspond to curves belonging to an infinite-dimensional space, and this evolution is modeled via functions. FDA is a fruitful line of research with applications in various domains, such as spectrometry, meteorology, physical and chemical processes, customer segmentation, or speech recognition [7, 8, 65, 67, 74]. Theoretically, functional data are assumed to be infinite-dimensional. In practice, such data are measured only on a (large) grid of points, which represents, for instance, the time instants. Because of their high dimensionality, functional data can be analyzed with the standard multivariate analysis techniques. Nevertheless, the direct use of such methodologies may have dramatic consequences, since the strong relationship between the measurements in two consecutive time instants is not taken into account, and limitations, such as the curse of dimensionality, may appear.

Consequently, many multivariate data analysis techniques have been developed in the FDA context, e.g. Principal Component Analysis (PCA) [12, 44], classification [56, 74], clustering [26, 57], or regression [17, 27].

Most studies on FDA have been focused on the univariate case, whereas the multivariate counterpart has received little attention. A multivariate functional datum is represented by a finite-dimensional vector where each covariate is defined by a different function. Moreover, the contributions on this topic are mainly devoted to PCA [4, 22, 46], and clustering [49, 52, 81], although we can also emphasize the recent work of [11] in the classification area. In this paper, we focus on a particular type of multivariate functional data, called hybrid functional data. They are finite-dimensional vectors that

✉ Asunción Jiménez-Cordero
asuncionjc@uma.es

Sebastián Maldonado
sebastianm@fen.uchile.cl

[1] Group OASYS. Ada Byron Research Building, C/ Arquitecto Francisco Peñalosa, 18, 29010, University of Málaga, Málaga, Spain

[2] Department of Management Control and Information Systems, School of Economics and Business, University of Chile, Santiago, Chile

[3] Instituto Sistemas Complejos de Ingeniería (ISCI), Santiago, Chile

combine static and functional features. By static features, we mean real or scalar covariates, whereas a functional feature is simply a function. We can find a plethora of examples of hybrid functional data in real life. For instance, in the field of medicine, functional features of a patient as the temperature or the electrocardiogram can be recorded, but also static variables, as the gender or the age. Despite its obvious application in real-world problems, this type of data has not been studied deeply in the literature. In fact, to the best of our knowledge, hybrid functional data have been analyzed only in [35] to select the most informative variables in terms of prediction in a real data application coming from the Spanish Energy Market, and in Chapter 10 of [72] where this type of data is sketched in a PCA context.

In this article, we are interested in classifying the hybrid functional data into two predefined classes. Functional data classification has been deeply studied in the literature. Although the standard multivariate classification methods can be applied to the functional context, some differences, such as the non-inversion of the covariance operator, are to be mentioned. The authors of [50] explain different methodologies to overcome this issue. On the other hand, the *near perfect classification* phenomenon only takes place in the functional context, as is detailed in [28]. Different classification methods has been developed, e.g. Partial Least Squares [70] or logistic regression, [73]. A survey with different strategies for classification methods in functional data can be seen in [3], whereas [66] presents some representations of functional data in classification. In this paper, we use the well-known technique Support Vector Machine (SVM). It has gained popularity due to its numerous virtues: the ability to construct nonlinear functions thanks to the Kernel Trick, its superior predictive performance compared to traditional parametric techniques, such as logistic regression, and the flexibility that allows its quadratic programming (QP) formulation [2, 84]. It has been applied widely in finite-dimensional data, e.g. [19, 24, 25, 62, 64]. Functional data classification with SVM has been discussed in several works in the literature. The first contributions on this topic are done in [74, 75]. There are some articles which focus on their interpretability [65] or their representation [67]. To see recent works on the topic, the reader is referred to [9, 11]. The SVM extension to hybrid functional data is discussed in Section 2.1. Feature selection is a key preprocessing step in data mining. A large number of covariates are usually associated with a lower value of the classification rate, due to the redundant information that they introduce. Furthermore, we should emphasize that the model is more interpretable if the number of variables is reduced. Hence, it is crucial to design a methodology which selects the most important features in terms of classification performance.

One of the issues related to kernel-based SVM classification is that the method is unable to derive the relevance of the variables automatically, constructing models using all available information [42, 61, 62]. Several feature selection strategies have been proposed to overcome this problem. Specifically, filter methods aim to select the most relevant features by ranking the covariates according to a metric. These methods are usually very fast since they do not take into account the training model. For instance, Fisher Score [32], measures the existing relationship between a single explanatory variable and the label vector, through the associated features that are then ranked according to this measure. An alternative type of feature selection approaches are wrapper methods. They measure the relevance of the features based on the classifier performance. The Recursive Feature Elimination SVM (SVM-RFE) [31, 41] is one of the most used wrapper methods applied in static feature selection. It removes those features whose removal leads to the largest margin of class separation in a backward fashion. Finally, embedded methods aim at determining a subset of relevant attributes during the classifier construction, encouraging sparsity via feature regularization, as done for example with the Lasso approach [14] which seeks an adequate balance between sparsity and predictive performance by replacing the Euclidean norm in the SVM formulation with the ℓ_1-norm.

Variable selection has also been applied in the univariate functional data field in studies such as [5, 82]. Nevertheless, in these cases, the variables are represented by the time instants during which the functions are measured. We also highlight the work of [39] in which functions are summarized in a set of features containing the maximum possible information, and then the most relevant covariates are selected with multivariate data analysis techniques.

To sum up, the contributions and objectives achieved in this paper are:

- We propose a new embedded feature selection method with a modification of the standard SVM-classification to handle functional hybrid data sets, and as a byproduct, selects the most informative features.
- We empirically demonstrate that such hybrid data sets cannot be learned properly with the current methodologies for SVM classification, due to the few number of references regarding feature selection in multivariate functional data and, more specifically, in hybrid functional data is very scarce.
- The proposed method allows weighting the different natures of the data, functional and static, by means of the scaling factors of a modified Gaussian kernel. The idea of considering different bandwidth values for different features is not new. Indeed, it has been

applied in [15, 21, 33, 76] for kernel density estimation purposes and in [63] for clustering problems.

The remainder of this paper is structured as follows: in Section 2 we formally describe the concepts used in our methodology and give the details of our approach. Section 3 is devoted to the computational experience. It includes the sensitivity analysis of our proposal given in Appendix A, as well as extra performance metrics, apart from the accuracy, namely the sensitivity, the specificity, and the Area Under Curve in Appendix B. Finally, the conclusions and possible future lines of research are described in Section 4.

2 The mathematical model

This section details the problem formulation of feature selection in SVM-classification with hybrid functional data. First, in Section 2.1 the main concepts of SVM for pure multivariate functional data are explained. Next, Section 2.2 is devoted to the extension of SVM to hybrid functional data, as well as to the problem formulation and the solving strategy.

2.1 Support vector machines for multivariate functional data classification

Let s be a sample of individuals with an associated pair (X_i, Y_i), $i \in s$. The datum $X_i \in \mathcal{F}^p$, is formed by a set of p functional features, i.e., $X_i = \left(X_i^1(t), \ldots, X_i^p(t)\right)$, where $X_i^v : [0, T] \to \mathbb{R}$, $v = 1, \ldots, p$ are functions belonging to the set $\mathcal{F}$ of Riemann integrable functions in the interval $[0, T]$. Moreover, $Y_i \in \{-1, +1\}$ denotes the class label of the observation i.

The benchmark SVM methodology [25], builds a hyperplane yielding a classification rule. The dual formulation of the SVM problem is stated as follows:

$$\begin{cases} \max\limits_{\alpha} \ \sum\limits_{i \in s} \alpha_i - \dfrac{1}{2} \sum\limits_{i,j \in s} \alpha_i \alpha_j Y_i Y_j K(X_i, X_j) \\ \text{s.t.} \ \sum\limits_{i \in s} \alpha_i Y_i = 0 \\ \qquad \alpha_i \in [0, C], \ i \in s, \end{cases} \tag{1}$$

where $C > 0$ is a regularization parameter, and $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is the so-called kernel function. As the decision rule: a new observation $X \in \mathcal{X}$ is assigned to class $+1$ if and only if $\hat{Y}(X) > \beta$, with β being a given threshold value. Here $\hat{Y}(X)$ is the score function, given by

$$\hat{Y}(X) = \sum_{i \in s} \alpha_i Y_i K(X, X_i), \qquad X \in \mathcal{X}, \tag{2}$$

One of the most used kernel functions, as reported in the literature, is the Gaussian kernel. It has been applied widely when finite-dimensional data are considered [18, 25, 54]. The extension to the functional case has also been studied. Indeed, the functional isotropic Gaussian kernel is analyzed in studies where univariate data appear [51, 67, 74, 75] and also in references dealing with multivariate data [85]. The expression of the isotropic Gaussian kernel for multivariate functional data, i.e. $X \in \mathcal{F}^p$, can be seen in (3):

$$K(X_i, X_j) = \exp\left(-\omega \sum_{v=1}^{p} \int_0^T \left(X_i^v(t) - X_j^v(t)\right)^2 dt\right) \tag{3}$$

for a single bandwidth ω which weighs all the covariates equally. Section 2.2 formally defines the hybrid functional data, describes how the kernel in (3) is extended to such type of data, and explains the proposed formulation for SVM classification and feature selection with hybrid functional data.

2.2 Problem formulation

An hybrid functional datum $X_i \in \mathcal{X}$, with $\mathcal{X} = \mathcal{F}^p \times \mathbb{R}^q$, is defined as a vector of p functional features and q static features. In other words, $X_i = (X_i^1(t), \ldots, X_i^p(t), X_i^{p+1}, \ldots, X_i^{p+q})$, where $X_i^v : [0, T] \to \mathbb{R}$, $v = 1, \ldots, p$ are functions belonging to the set $\mathcal{F}$ of Riemann integrable functions in the interval $[0, T]$, and $X_i^v \in \mathbb{R}$, $v = p + 1, \ldots, p + q$.

The main objective of this paper is to design a model which obtains, via SVM, good classification rates in order to determine the class $Y \in \{-1, +1\}$ of a new observation $X \in \mathcal{X}$, at the same time that it yields the most informative set of features $\mathcal{V} \subset \{1, \ldots, p + q\}$. To do this, we modify the standard Gaussian functional kernel in (3), in which a single bandwidth is considered, by associating a bandwidth with each feature, yielding the following expression:

$$K(X_i, X_j, \boldsymbol{\omega}) = \exp\left(-\sum_{v=1}^{p} \omega_v \int_0^T \left(X_i^v(t) - X_j^v(t)\right)^2 dt - \sum_{v=p+1}^{p+q} \omega_v (X_i^v - X_j^v)^2\right), \tag{4}$$

for $X_i, X_j \in \mathcal{X}$. Notice that the dependency of the bandwidth $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_{p+q})$ on the kernel K is highlighted through the notation $K(X_i, X_j, \boldsymbol{\omega})$.

Our proposed kernel in (4) differs from the kernel in (3) in the role that the bandwidth plays. Whereas the bandwidth in (3) is just a single value, common to all the variables, the kernel in (4) has a bandwidth for each feature, which allows more flexibility in our model, weighting each covariate differently according to its contribution in the classification model, and allowing the link between variables of different nature, static and functional.

The feature selection problem implies the tuning of two parameters: the regularization parameter C of the SVM problem (1), and the bandwidths $\omega_v, v = 1, \ldots, p + q$ associated with each feature of $X \in \mathcal{X}$ through the kernel (4).

In agreement with the methodologies of [9, 11], we propose combining a grid search to get the optimal value of C with a bilevel optimization problem which will yield the optimal bandwidth $\boldsymbol{\omega}$.

Multiple criteria can be used in the objective function of the bilevel optimization problem. Minimizing the misclassification rate is the usual approach utilized. Nevertheless, such a choice is a linear piecewise function which prevents the use of gradient-based optimization searches. We propose, instead, defining the objective function as the maximization of the Pearson correlation, ρ, between the class label Y_i and the score $\hat{Y}(X_i, \boldsymbol{\omega}, \alpha)$ in (2). The Pearson correlation coefficient has been used before in [9, 11] as surrogate of the number of misclassified with outstanding results. Although we are defining a linear relationship between vectors of different nature, since Y is a binary vector taking values in $\{-1, +1\}$ and $\hat{Y}$ is a real vector; the numerical experience in [9, 11] has shown that the usage of the Pearson correlation has two big advantages. On the one hand, this coefficient is very easy and fast to compute. On the other hand, the continuous behavior allows one to apply smooth optimization strategies. Parameter tuning usually leads to overfitting when the whole data set is considered. To avoid this issue, we divided the sample s into three independent parts, s_1, s_2 and s_3. Sample s_1 is utilized to solve the SVM problem (1), for fixed C and $\boldsymbol{\omega}$, yielding the variables α. The independent sample s_2 is used to measure the goodness of fit via the correlation $\rho((Y_i, \hat{Y}(X_i, \boldsymbol{\omega}, \alpha))_{i \in s_2})$ for α and C fixed. Finally, sample s_3 is employed to find the regularization parameter C, by computing the accuracy on s_3 for a given C in the grid, and keeping the one with the largest value. Therefore, for a fixed C, the bilevel optimization problem is stated as follows:

$$\left\{\begin{array}{ll} \max\limits_{\boldsymbol{\omega}, \alpha} & \rho((Y_i, \hat{Y}(X_i, \boldsymbol{\omega}, \alpha))_{i \in s_2}) \\ \text{s.t.} & \alpha \text{ solves (1) in } s_1 \\ & \omega_v \geq 0, \quad \forall v, \end{array}\right. \tag{5}$$

Nonlinear bilevel optimization problems, such as (5), can be solved with the off-the-shelf methodologies described in [23]. Nevertheless, such strategies are computationally expensive. We propose using an alternating approach instead, as was done in [9, 11].

Our alternating approach consists of just a few iterations of two steps. First, Problem (1) is solved, for fixed $\boldsymbol{\omega}$ in sample s_1, yielding the optimal variables α. Secondly, for fixed α, Problem (6) is solved in sample s_2, giving the optimal values of the parameter $\boldsymbol{\omega}$.

$$\left\{\begin{array}{ll} \max\limits_{\boldsymbol{\omega}} & \rho((Y_i, \hat{Y}(X_i, \boldsymbol{\omega}))_{i \in s_2}) \\ \text{s.t.} & \omega_v \geq 0, \quad \forall v, \end{array}\right. \tag{6}$$

Problems (1) and (6) have different natures and, consequently, they should be solved with different strategies. Problem (1) is a quadratic maximization problem with linear constraints in which SMO-like algorithms can be applied to easily reach the global optimum of the problem. In contrast, Problem (6) is a continuous optimization problem whose optimal solution is obtained by combining classic local searches and a multi-start approach.

The alternating procedure is run, for a fixed C, until some stopping criterion is reached. Notice that, apart from obtaining good classification rates, our goal is to select the most informative features. To do this, once the alternating approach is finished, we eliminate those covariates v whose associated bandwidths ω_v are close enough to zero, and repeat the alternating algorithm with the remaining features. In other words, we keep those features satisfying $\omega_v > \delta$, where $\delta > 0$ is a threshold value. This process is repeated until the selected features do not change in two consecutive iterations.

Once the alternating approach provides good values for α, $\boldsymbol{\omega}$, and therefore, the set $\mathcal{V}$ of selected features, the value of C is chosen by computing the accuracy on s_3 for all C values in the grid, and the one that leads to the largest accuracy is kept.

Finally, the effectiveness of our methodology is tested on an independent sample s_4, in which the classification accuracy is computed. A pseudocode of our approach is given in Algorithm 1.

Algorithm 1 Heuristic for parameter tuning.

```
• Randomly split the sample $s$ into $s_1$, $s_2$, $s_3$ and $s_4$.
for $C$ in the grid do
    Initialization: $\mathcal{V} = \{1, \ldots, p + q\}$
    repeat
        Alternating Procedure
        repeat
            1. For $\boldsymbol{\omega}$ fixed, obtain the variables $\alpha$ of the
               SVM classifier by solving Problem (1) in $s_1$.
            2. For a fixed $\alpha$, calculate $\boldsymbol{\omega}$ by solving
               Problem (6) in $s_2$.
        until stopping criterion
        • Delete the features, $v$, such that $\omega_v \leq \delta$, i.e.
          $\mathcal{V} = \{v : \omega_v > \delta\}$
    until no new features are deleted
    • Evaluate the accuracy in the sample $s_3$ with $C$ fixed.
end for
• Keep the value of $C$ with the maximum accuracy in $s_3$,
and the associated values of $\alpha$, $\boldsymbol{\omega}$, and the set $\mathcal{V}$.
Output: optimal parameters $C$ and $\boldsymbol{\omega}$, optimal classification coefficients $\alpha$, the selected features in $\mathcal{V}$, and the
corresponding accuracy estimated from $s_4$.
```

Table 1 Data description summary (including number of individuals and records of each label)

Data set	# individuals	# records label -1	# records label $+1$
batch	1000	500	500
trigonometric	1000	500	500
pen	296	171	125
retail	3602	1176	1826

3 Numerical Experiments

This section is devoted to the computational experience. In Section 3.1, the different databases are explained. Section 3.2 is devoted to the description of the experiments performed. Section 3.3 details the approaches utilized to compare our algorithm with. Finally, Section 3.4 gives the results of our proposal, including the sensitivity analysis explained in Appendix A.

3.1 Data Set Description

Two simulated examples, namely *batch* and *trigonometric*, and two real databases, denoted here as *pen* and *retail*, were studied. A summarized description of the data sets, including the number of individuals in the sample, the number of elements of each class, and the number of static and functional covariates as well as their names, can be seen in Tables 1 and 2.

Sections 3.1.1–3.1.4 detail how the different databases have been generated and Figs. 1, 2, 3 and 4 show respectively a subset of ten functions of the data sets *batch*, *trigonometric*, *pen* and *retail*. The functional features are depicted in a standard $x - y$ plot, where the solid blue lines indicate the individuals with class 1 and the dashed red line mark the observations with class -1. On the other hand, for the sake of visualization static covariates are shown in boxplots (or barplots in the case of categorical features), with the individuals with classes 1 and -1 colored in blue and red respectively.

3.1.1 Batch data set

The three functional covariates of the first data set, *batch*, come from Section 4.1 of Wang and Yao [85]. Although Wang and Yao [85] consider that the upper bound for the time interval in which the functions are measured follows an uniform distribution on $[0.9, 1.1]$, we assume, for the sake of simplicity, that $X^v : [0, 1] \to \mathbb{R}$, $v = 1, 2, 3$. Formally:

$$\begin{aligned} X_i^1(t) &= a_i \cdot t + \varepsilon_i^1(t) \\ X_i^2(t) &= a_i \cdot t^2 + \varepsilon_i^2(t) \\ X_i^3(t) &= b_i\,(4\sin(t) + 0.5\sin(\nu_0 \cdot t)) \end{aligned}$$

for $t \in [0, 1]$, where (a_i, b_i) follows a bivariate Gaussian distribution with mean vector $(2.5, 2.5)$ and covariance matrix $diag(2.5, 2.5)$.

For each $t \in [0, 1]$, the measurements errors $\varepsilon_i^1(t)$ and $\varepsilon_i^2(t)$ are i.i.d. Gaussian noise with mean 0 and standard deviation 0.2. The individuals X_i with label $Y_i = 1$ have $\nu_0 = 10$, whereas those with $Y_i = -1$ are associated with $\nu_0 = 11$.

Therefore, the third covariate is the only one that is relevant for classification, if just the functional component of the hybrid functional data is taken into account.

To complete the data set, we added two real variables, X^4 and X^5, in agreement with (7) and (8) for all $i = 1, \ldots, 1000$:

$$X_i^4 \sim \begin{cases} \mathcal{N}(\mu = 39, \sigma = 1), \text{ if } Y_i = 1 \\ \\ \mathcal{N}(\mu = 40, \sigma = 1), \text{ if } Y_i = -1 \end{cases} \tag{7}$$

$$X_i^5 \sim \begin{cases} \mathcal{N}(\mu = 2, \sigma = 1), \text{ if } Y_i = 1 \\ \\ \mathcal{N}(\mu = 3, \sigma = 1), \text{ if } Y_i = -1 \end{cases} \tag{8}$$

where $\mathcal{N}(\mu, \sigma)$ indicates a normal distribution of mean μ and standard deviation σ.

Table 2 Data description summary (including number of features and their names)

Data set	# functional covariates	Name functional covariates	# static covariates	Name static covariates
batch	3	1, 2 and 3	2	4 and 5
trigonometric	2	1 and 2	2	3 and 4
pen	2	x trajectory and y trajectory	1	force
retail	5	Amount, Quantity, Recency, Frequency and Monetary	1	UK Customer?

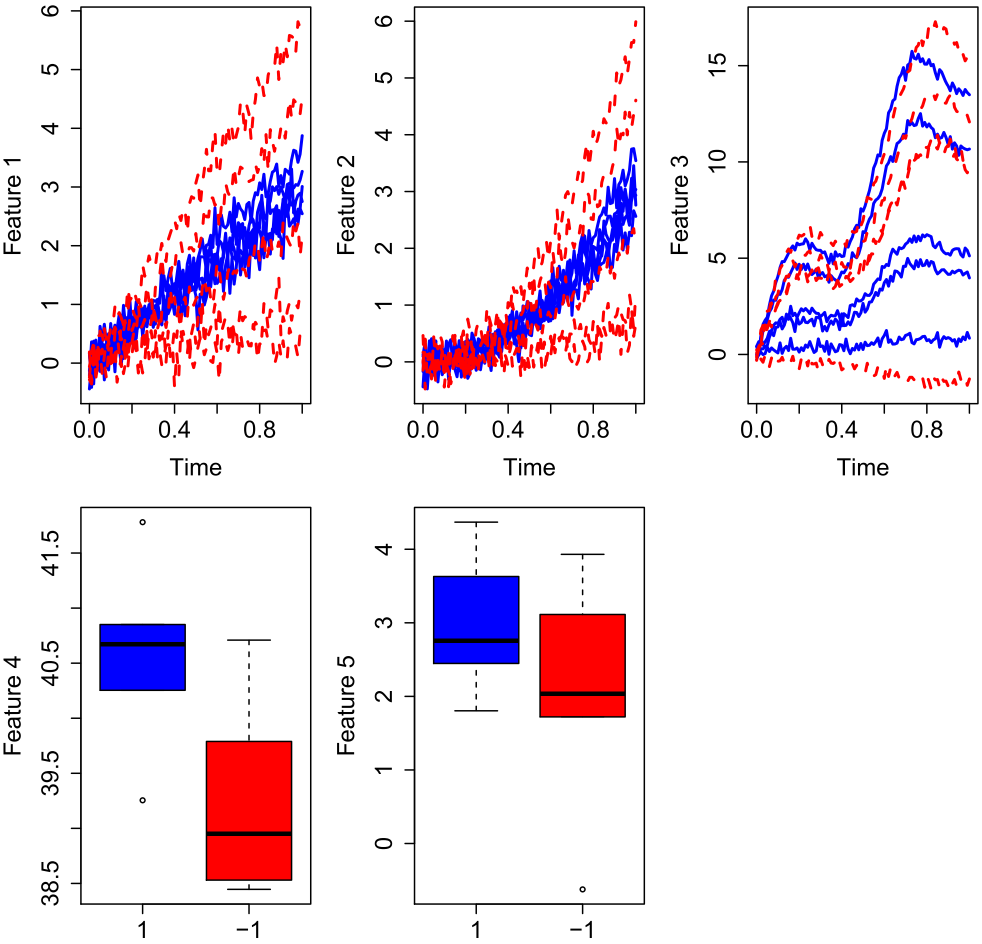


Fig. 1 Subset of *batch* data set

3.1.2 Trigonometric data set

The *trigonometric* database is formed by two functional features and two scalar covariates. Functional components $X_i^v : [1, 21] \longrightarrow \mathbb{R}$, $v = 1, 2$ are based on the data generated in Section 5.2.2 of [49] and have the form:

$$X_i^1(t) = -\frac{21}{2} + t + \nu_0 U_1 \cos\left(\nu_0 \frac{t}{10}\right) + \nu_0 U_1 \sin\left(\nu_0 + \frac{t}{10}\right) + \varepsilon_i^1(t)$$

$$X_i^2(t) = -\frac{21}{2} + t + \nu_0 U_1 \sin\left(\nu_0 \frac{t}{10}\right) + \nu_0 U_2 \cos\left(\nu_0 + \frac{t}{10}\right) + \nu_0 U_3 \left(\left(\frac{t}{10}\right)^2 + \frac{t}{10} + 1\right) + \varepsilon_i^2(t)$$

where $t \in [1, 21]$, $U_1, U_2, U_3 \sim \mathcal{N}(1, 1)$ are independent Gaussian variables and $\varepsilon_i^1(t)$ and $\varepsilon_i^2(t)$ are white noise of unit standard deviation.

The value of ν_0 is dependent on the class label. More specifically, the individuals with label $Y_i = 1$ have $\nu_0 = 1$, while the observations corresponding to $Y_i = -1$ have $\nu_0 = 2$.

The remaining static variables X^3 and X^4 have been created according to (9) and (10)

$$X_i^3 \sim \begin{cases} \mathcal{N}(\mu = 0, \sigma = 15), & \text{if } Y_i = 1 \\ \mathcal{N}(\mu = 20, \sigma = 20), & \text{if } Y_i = -1 \end{cases} \tag{9}$$

$$X_i^4 \sim \mathcal{N}(\mu = 0, \sigma = 1), \quad \forall i \tag{10}$$

3.1.3 Pen data set

The *pen* data set comes from the *Character Trajectories data set* of the UCI Machine Learning repository [30] and has been used in papers such as [47, 48]. It contains the x and y trajectories, and the force with which multiple characters have been written. This data set is usually applied in multiclass classification frameworks, e.g., [77, 78, 80], where the labels of 20 characters are to be predicted. Since

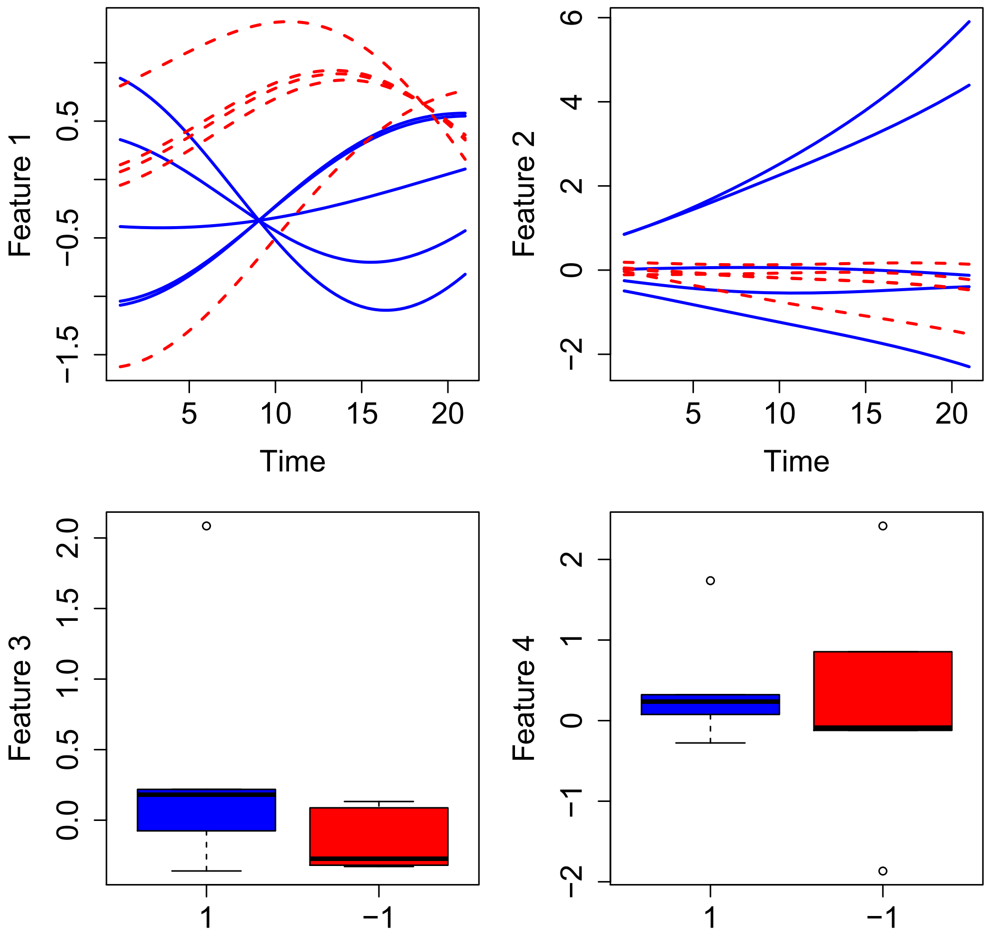


Fig. 2 Subset of *trigonometric* data set

in this paper, we focus on a binary classification problem, we have adapted this data set to our setting. In particular, our aim is to classify between two randomly selected characters. In this case, we have chosen to distinguish between m and z, corresponding to the class labels 1 and -1, respectively. The two functional features here considered are the x and y trajectories, while the pen tip force is the static covariate.

3.1.4 Retail data set

The second real-world application database *retail* is extracted from the *Online Retail Data Set* of the UCI Machine Learning Repository [30] and has been studied in [20]. It contains the monthly transactions of the customers of a UK-registered non-store, online retail during the first 10 months of the 13 months available. This database is originally used for clustering problems, where the customers are to be grouped according to their monthly transactions. In this paper, we focus on binary classification, and therefore the original database has been conveniently modified. Indeed, here the aim is to predict whether the customer will buy products in the last three months. Customers that only purchased items in the last three months were removed from the data set since no purchase history is available for constructing covariates, yielding an amount of 3,602 individuals instead of the original number of 3,630. The first functional feature is the amount of money spent by the customers. The second functional variable denotes the quantity of products bought. The last three functional covariates are the variables Recency, Frequency, and Monetary described in [20]. Finally, the scalar variable is a binary feature that indicates whether the customers come from the UK, coded by 1, or not, coded by 0.

3.2 Description of the experiments

This section explains the details of the computational experiments carried out to show the efficiency of our approach. Algorithm 1 has been run on the databases described in Section 3.1. Each data set is split into four parts, $s_1 - s_4$, whose roles are explained in Section 2.2. Since the features of the hybrid functional data may have different scales, we have normalized separately each feature before performing our approach, as explained in [85]. When selecting the most informative covariates, we remove those features such that $\omega_v \leq 10^{-5}$, i.e. $\delta = 10^{-5}$. The stopping

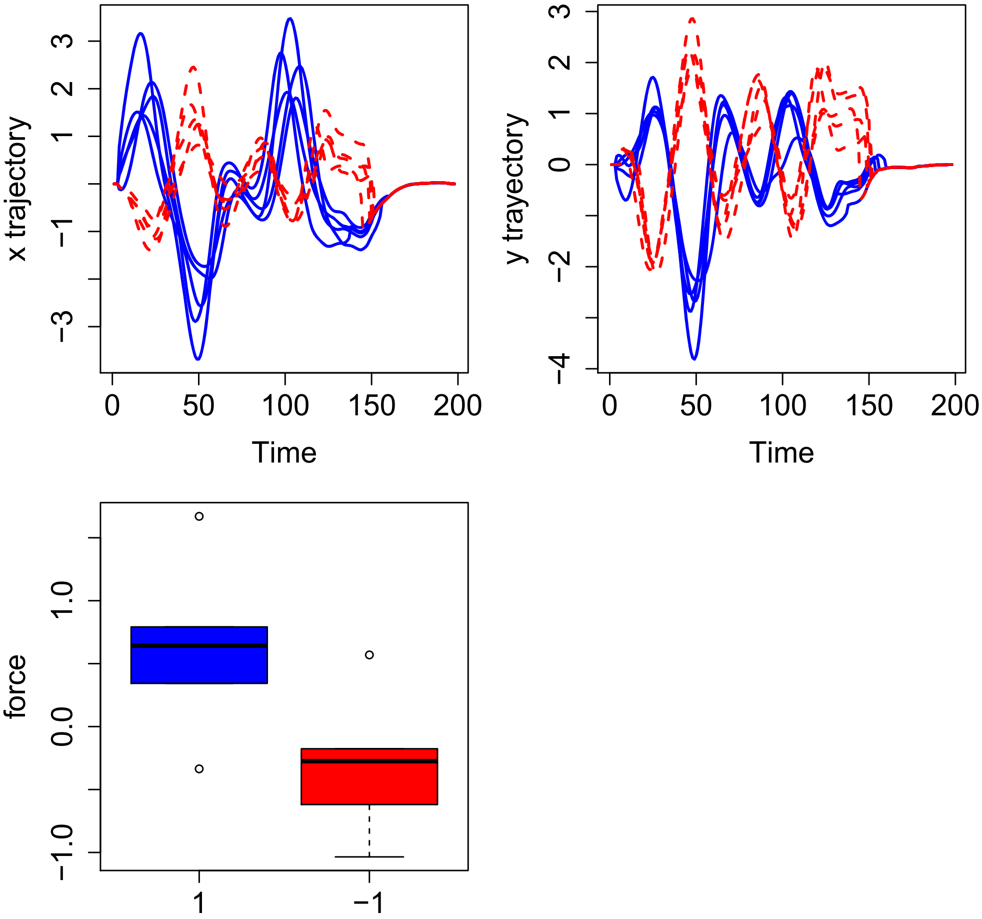


Fig. 3 Subset of *pen* data set

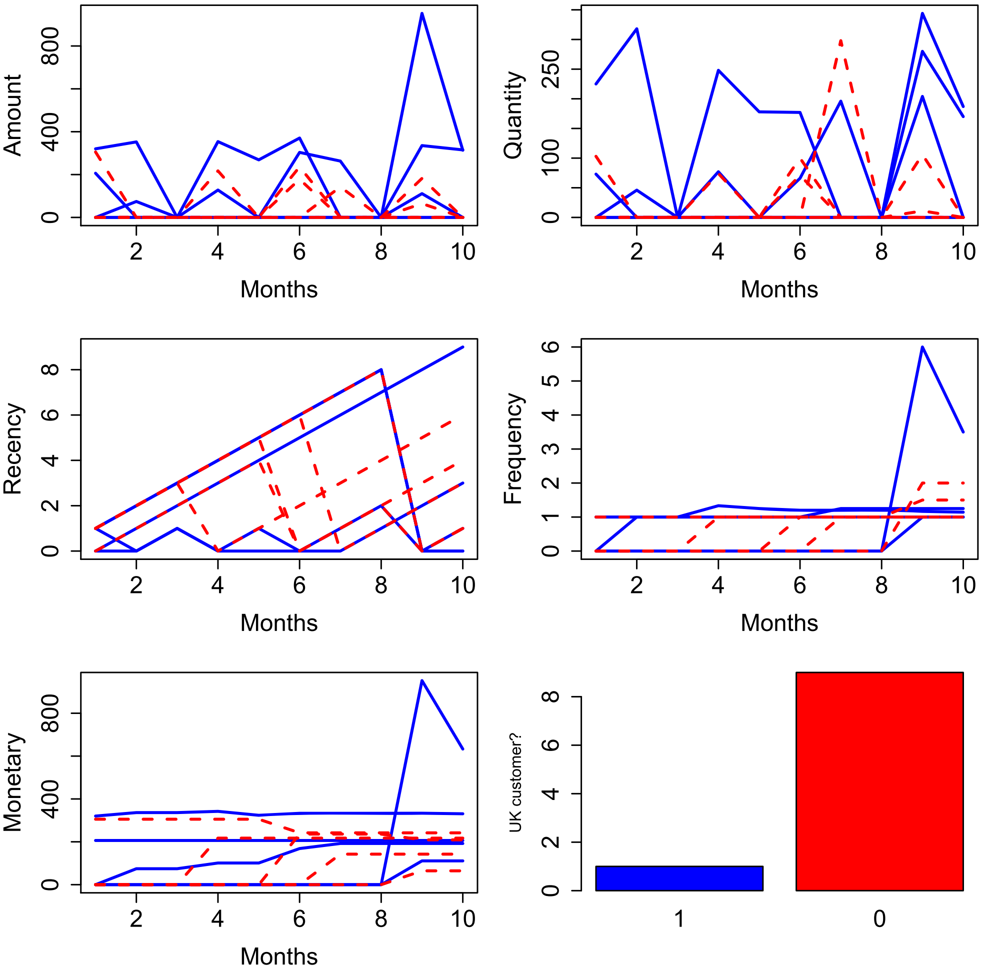
criterion is reached when the number of iterations is equal to five or when the values of the bandwidths, and therefore the selected features, do not change in two consecutive iterations. The parameter C moves in the set $\{2^{-7}, \ldots, 2^7\}$ on a logarithmic scale. In order to have stable results, Algorithm 1 was run five times, and the average accuracy on the test sample s_4 is reported in Table 3. To compare our methodology with others, we consider the approaches detailed in Section 3.3 on the normalized data sets. The average accuracy of the comparative methods on the very same test sample is also given in Table 3.

In order to confirm our results, we perform the Friedman and Holm tests to evaluate the statistical significance, widely applied in the literature in papers such as [37, 59]. These tests were proposed in [29] to compare various machine learning strategies on multiple data sets. Firstly, the average rank is calculated for our approach and for all the alternative algorithms based on the accuracy of all data sets. Secondly, the Friedman test is applied for the hypothesis test which checks whether all the algorithms are equivalent or not in terms of performance. If the null hypothesis of similar performance is rejected, then the Holm post-hoc test is applied for pairwise comparisons between the algorithm with the highest rank and the rest of them. Each hypothesis test assesses whether the average accuracy of the algorithm with the highest rank and the comparative methodology is equal or not. The resulting $p-$values are sorted in increasing order, and the null hypothesis is rejected if the $p-$value is below a fixed significance threshold. In all these tests, we use $\alpha = 0.05$ as significance level. Furthermore, we executed a sensitivity analysis in order to study the accuracy with respect to the parameters involved in the algorithm. The details of this analysis are explained in Appendix A. All the experiments were coded in R, [79], and carried out in a cluster with 2 terabytes of RAM memory at 6.2 TFlops, running CentOS Linux 7.3.

3.3 Comparative algorithms

Since, to the best of our knowledge, no methodology has been reported in the literature that deals with feature selection in hybrid functional data, we suggest some techniques with which to compare our proposal, even though not all of them are able to perform feature selection. Notice that the main objective of our approach is to obtain good classification rates at the same time that we select the most important features. The first algorithm gives the results of the classification of the hybrid functional data

Fig. 4 Subset of *retail* data set



when no feature selection is made. The second comparative method treats the functional component of the hybrid functional data as static by summarizing the functions into a finite-dimensional vector. Such static extraction is done in two different ways. On the one hand, we summarize each functional component into a 4−dimensional vector including the mean value, the standard deviation, the maximum and the minimum values. On the other hand, each functional covariate is considered as a finite-dimensional vector whose components are the evaluation of the functions in the discretization time points where they have been actually measured. We also compare our proposal with the eight regularized classification methods which can be found on the R library `LiblineaR`. Particularly, we have applied the eight classification regularization schemes they provided on the discretized hybrid functional data. Finally, we include the comparison of our approach with six filter methods included in the R library `mlr` on the discretized hybrid functional data. In all the above-explained algorithms the data set is divided into three parts, namely, training, validation, and test. For the sake of comparison with our proposed approach, the division is made in such a way that the test sample coincides exactly with the so-called sample s_4 described in Section 2.2. Furthermore, all the comparative algorithms were run five times for each data set, as stated in Section 3.2. The accuracy over all the runs, measured on the test sample, is used as the performance metric, and is given in Table 3. Sections 3.3.1–3.3.3 give details about all the comparative methods.

3.3.1 Functional SVM (FSVM)

The first alternative method corresponds to the SVM algorithm for functional data. In this case, the different types of features are not taken into account, and no variable selection is made. A grid search is performed to obtain the scalar parameters C and ω based on the following set of values: $\{2^{-7}, \ldots, 2^{7}\}$ on a logarithmic scale. The SVM problem (1) is run with an isotropic Gaussian kernel in (11):

$$K(X_i, X_j) = \exp\left(-\omega\left(\sum_{v=1}^{p}\int_0^T\left(X_i^v(t) - X_j^v(t)\right)^2 dt + \sum_{v=p+1}^{q}\left(X_i^v - X_j^v\right)^2\right)\right) \tag{11}$$

Table 3 Result summary

Data set	batch	trigonometric	pen	retail
Alt. appr.	**95.68**	**98.64**	**99.20**	64.06
FSVM	74.08	96.32	**99.20**	62.73
ℓ_2-SVM (4 dim)	76.72	83.76	98.93	62.08
ℓ_2-SVM (disc)	90.48	96.00	98.93	63.83
ℓ_2-LR$_p$	95.20	90.88	**99.20**	66.70
$\ell_2\ell_2$-SVM$_d$	92.88	88.08	**99.20**	60.70
$\ell_2\ell_2$-SVM$_p$	93.84	91.20	**99.20**	67.06
$\ell_2\ell_1$-SVM$_d$	92.80	91.12	**99.20**	63.98
SVM$_{CS}$	92.72	80.96	**99.20**	66.65
$\ell_1\ell_2$-SVM	93.68	92.64	**99.20**	67.00
ℓ_1-LR	93.76	92.32	**99.20**	66.96
ℓ_2-LR$_d$	93.44	92.16	**99.20**	**67.30**
$\mathcal{X}^2$ test	54.32	98.08	98.93	63.28
information_gain	54.32	98.40	98.93	63.28
kruskal_test	55.04	98.32	98.93	64.03
min_depth	51.68	98.40	**99.20**	63.51
rf_importance	52.32	98.48	98.67	63.90
variance	53.52	98.24	**99.20**	63.28

Accuracy as performance measure. For each data set, we have highlighted in bold the highest accuracy value among all the methods. The average accuracy on the test sample for all the approaches is given

for $X_i, X_j \in \mathcal{X}$. The parameters C and ω that lead to the best results in terms of the classification rate on the validation sample are kept. Finally, the accuracy of the selected parameters C and ω is computed as a measure of performance.

3.3.2 Standard (static) SVM (ℓ_2-SVM)

The second alternative approach corresponds to the soft-margin SVM model [24] when the functions of the hybrid functional data are summarized in scalar values. We solved the SVM problem (1) on the training set, for each of the values of C and ω belonging to the set $\{2^{-7}, \dots, 2^7\}$ in logarithmic scale.

In this case, the kernel function used in Problem (1) is the isotropic kernel in (12) for multivariate data, in which a transformation of X_i, namely Z_i, is used:

$$K(Z_i, Z_j) = \exp\left(-\omega \|Z_i - Z_j\|^2\right), \tag{12}$$

where $\|\cdot\|$ denotes the ℓ_2-norm.

The best values of C and ω are chosen by measuring the accuracy on the validation sample, and then, the final results are estimated with the optimal values for C and ω on the test sample.

Two different transformations Z_i are here suggested. In the first one, each functional component $X_i^v(t)$, $v = 1, \dots, p$ is summarized in a 4−dimensional vector which includes the mean value, the standard deviation, the minimum and the maximum values. Moreover, we add the values of the static covariates X_i^v, $v = p+1, \dots, p+q$. Such transformation Z_i is given in (13):

$$\begin{aligned} Z_i = \Big(&\text{mean}(X_i^1(t)),\ \text{sd}(X_i^1(t)),\ \min(X_i^1(t)),\ \max(X_i^1(t)), \dots, \\ &\text{mean}(X_i^p(t)),\ \text{sd}(X_i^p(t)),\ \min(X_i^p(t)),\ \max(X_i^p(t)), \\ &X_i^{p+1}, \dots, X_i^{p+q}\Big) \end{aligned} \tag{13}$$

The second transformation here proposed consists of substituting each functional covariate by the H discretization points, $t_1, \dots, t_H$, where it has been recorded. We also add the values of the static covariates. In other words, the transformation Z_i turns out to be as in (14):

$$Z_i = \left(X_i^1(t_1), \dots, X_i^1(t_H), \dots, X_i^p(t_1), \dots, X_i^p(t_H), X_i^{p+1}, \dots, X_i^{p+q}\right) \tag{14}$$

3.3.3 Regularized classification methods

We also compare our proposal with eight regularized algorithms in order to assess the performance of various feature selection strategies that has been used with SVM classification in recent studies (see e.g. [1, 43, 58, 69, 86]). These eight methods stem from the well-known `LiblineaR` library [34]. The following strategies are studied:

- ℓ_2-regularized logistic regression, primal implementation (ℓ_2-LR$_p$).
- ℓ_2-regularized SVM with ℓ_2-norm loss function, dual implementation ($\ell_2\ell_2-$SVM$_d$).
- ℓ_2-regularized SVM with ℓ_2-norm loss function, primal implementation ($\ell_2\ell_2-$SVM$_p$).
- ℓ_2-regularized SVM with ℓ_1-norm loss function, dual implementation ($\ell_2\ell_1-$SVM$_d$).
- The SVM implementation by Cramer and Singer (SVM$_{CS}$).
- ℓ_1-regularized SVM with ℓ_2-norm loss function ($\ell_1\ell_2-$SVM).
- ℓ_1-regularized logistic regression (ℓ_1-LR).
- ℓ_2-regularized logistic regression, dual implementation (ℓ_2-LR$_d$).

For each regularized method, the functional covariates were transformed into static variables by using (14). The trade-off parameter C is sought in the set $\{2^{-7}, \dots, 2^7\}$ using a logarithmic scale, and the value yielding the best accuracy on the validation sample is saved. Finally, the accuracy of the best value of C is given as a result.

3.3.4 Filter methods

Finally, the proposed approach has been also compared with the following six filter methods provided by the recent R library mlr, [6, 13]:

- Chi-squared test ($\mathcal{X}^2$ test).
- Information gain entropy (information_gain).
- Kruskal-Wallis test (kruskal_test).
- Minimal depth variable selection (min_depth).
- Random forest variable importance (rf_importance).
- Low-variance method (variance).

These methodologies have been recently applied in works such as [16, 38, 40, 45, 53, 55, 60, 68, 83]. More precisely, the functional covariates have been transformed according to (14). Then, each of the above methods has been run on the transformed covariates and the 25% of the most relevant ones is selected. Such selected variables are used to train the SVM model (1) for a given C and applying different kernel functions. In particular, we have run the experiments using the standard multivariate Gaussian kernel with a fixed bandwidth, $\omega \in \{2^{-7}, \ldots, 2^7\}$, the polynomial kernel with degree parameter $d \in \{1, \ldots, 5\}$ and constant c in the set $\{-2, \ldots, 2\}$, and the sigmoid kernel with offset parameter ranging also in the set $\{-2, \ldots, 2\}$. The best values of C are found in the set $\{2^{-7}, \ldots, 2^7\}$ in logarithmic scale, and the value with the largest accuracy and the best kernel choice on the validation sample is kept. The final results collect the accuracy on the test set for the best kernel hyperparameters and the regularization parameter C.

3.4 Experimental results

Algorithm 1 and all the comparative methods of Section 3.3 have been run five times. Table 3 shows the average accuracy values on the test sample. For each data set, we have highlighted in bold the best algorithm which is associated with the highest accuracy. Moreover, our approach is denoted as *Alt. appr.*, and the FSVM strategy of Section 3.3.1 is designated with the very same name. The ℓ_2-SVM method for the finite-dimensional data in (13) and (14) are denoted as *ℓ_2-SVM (4 dim)* and *ℓ_2-SVM (disc)*, respectively. Finally, the accuracy results of the eight classification methodologies of LiblineaR in Section 3.3.3 are indicated by ℓ_2-LR$_p$, $\ell_2\ell_2$-SVM$_d$, $\ell_2\ell_2$-SVM$_p$, $\ell_2\ell_1$-SVM$_d$, SVM$_{CS}$, $\ell_1\ell_2$-SVM, ℓ_1-LR, ℓ_2-LR$_d$, whereas the accuracy given by the six filter methods of mlr detailed in Section 3.3.4 are denoted by $\mathcal{X}^2$ *test*, *information_gain*, *kruskal_test*, *min_depth*, *rf_importance* and *variance*.

As a general conclusion from Table 3 we can state that our strategy is the best one in data sets *batch* and *trigonometric*. In the *pen* data set, we obtain comparable results with the existing methods, whereas the *retail* database is slightly better classified with the ℓ_2-LR$_d$ strategy than with ours. More detailed information about the results is given in Sections 3.4.1–3.4.4.

The results obtained in Table 3 using accuracy as performance measure are complemented in Appendix B, in which we present the Area Under the Curve (AUC), sensitivity, and specificity metrics for all methods and data sets. These new metrics support the conclusions reported for Table 3, confirming that our proposal achieves the best predictive performance compared to the alternative classification techniques. In particular, our approach achieved the best sensitivity in all four data sets, the best specificity in two of the four data sets, and the best AUC in three of the four data sets. Furthermore, our proposal achieved competitive results in the data sets in which was not able to be the best-ranked method.

Apart from Table 3, we provide the average rank and the average accuracy of all the tested methods. For each methodology, the average rank is computed as the mean over all the ranks associated to the four databases. Such a rank is obtained by sorting in decreasing order the accuracy values. The average accuracy is simply obtained by computing the mean value over all the data sets of the accuracy results which appear in Table 3. It is clear that our approach is the best one when comparing with the remaining 17 methods. Indeed the average rank of the proposed methodology is 3.875 which is clearly far from the second and third best methods, $\ell_1\ell_2$SVM and $\ell_2\ell_2$SVM$_p$, both of them with an average rank of 6.125 (Table 4).

Table 4 Average rank and accuracy for all the methods

Method	Average rank	Average accuracy
Alt. appr.	3.8750	89.3950
$\ell_1\ell_2$-SVM	6.1250	88.1300
$\ell_2\ell_2$-SVM$_p$	6.1250	87.8250
ℓ_1-LR	6.3750	88.0600
ℓ_2-LR$_d$	6.3750	88.0250
ℓ_2-LR$_p$	7.1250	87.9950
$\ell_2\ell_1$-SVM$_d$	9.3750	86.7750
SVM$_{CS}$	9.8750	84.8825
min_depth	10.0000	78.1975
kruskal_test	10.2500	79.0800
FSVM	10.6250	83.0825
variance	10.6250	78.5600
ℓ_2-SVM (disc)	11.2500	87.3100
information_gain	11.7500	78.7325
rf_importance	11.7500	78.3425
$\ell_2\ell_2$-SVM$_d$	11.8750	85.2150
$\mathcal{X}^2$ test	12.6250	78.6525
$\mathcal{X}^2$ test	12.5000	77.8650
ℓ_2-SVM (4 dim)	15.0000	80.3725

3.4.1 Batch data set

If we observe Table 3, it is quite apparent that the proposed methodology yields better results. Furthermore, we are able to identify the most informative features as a byproduct. In fact, the third variable was selected to be important by our algorithm in all the five runs. Remember that this feature is the only functional covariate that is correlated with the target variable. In the third run, for instance, we obtain the following optimal bandwidth: $\boldsymbol{\omega} = (0, 0, 165.9076, 0.0703, 0)$, i.e. the third and the fourth variables are identified as relevant. Notice that our methodology is not influenced by the static or functional nature of the covariates. In fact, in this example, one variable of each type is selected.

Regarding the sensitivity analysis of the parameters (see Appendix A), we observe that the value of C should be carefully chosen since, as can be seen in Fig. 5, the resulting accuracy depends on the value of C.

By contrast, our proposal is robust with respect to the elimination threshold δ and the number of iterations of the alternating approach, as shown by the stable behavior in Figs. 5b and c, respectively.

Finally, in Fig. 6 we see how the optimal values of the bandwidths evolve in the five runs. We observe that independent of the initial bandwidths selected, the bandwidth associated with the third variable tends toward a value greater than zero.

3.4.2 Trigonometric data set

Table 3 shows that our proposal improves the performance measure of the comparative algorithms. With respect to the feature selection output, features one and three are selected in the five runs, and variable two in three out of five. Indeed, the fourth run gives $\boldsymbol{\omega} = (0.3758, 0.1281, 0.0929, 0)$ as optimal solution. Focusing on the sensitivity analysis with respect to δ and the number of iterations, we state that stability in the results is obtained. Nevertheless, the value of C has an important role in the accuracy values. See Fig. 7 in Appendix A for more details. The evolution of the values of the bandwidths in all the five runs is depicted in Fig. 8.

3.4.3 Pen data set

Focusing on Table 3 we observe that our methodology is comparable with the rest of the strategies. As it was sketched in Section 3.1.3, this database is usually applied for multiclass classification purposes. Even though the results are not comparable, we want to remark that the best accuracy results obtained in this data set for multiclass classification in [77, 78, 80] are 94.50%, 88% and 84.5%, respectively. Regarding the number of relevant features, we should say that our approach selects just one variable out of three in two of the five runs. The evolution of the bandwidths values can be seen in Fig. 10 of Appendix A. In this example, the value of C is a critical point as can be observed in Fig. 9c, since the difference between the best and the worst case is around 40 points. However, our method is robust with respect to δ and the number of iterations, as shown in Figs. 9b and c.

3.4.4 Retail data set

We observe in Table 3 that our proposal yields better results than the strategies *FSVM, ℓ_2-SVM (4 dim), ℓ_2-SVM (disc)*, *$\ell_2\ell_2$-SVM$_d$*, *$\ell_2\ell_1$-SVM$_d$*, *$\mathcal{X}^2$ test*, *information_gain*, *kruskal_test*, *min_depth*, *rf_importance* and *variance*, and slightly worse results than the remaining methodologies. Moreover, the selected variables are the third and sixth in four of five runs. As an illustration, the optimal bandwidth in one of these runs is $\boldsymbol{\omega} = (0, 0, 1.5887, 0, 0, 45.5919)$. Feature 3 and Feature 6 correspond to Recency (number of months since the last purchase) computed for each of the 10 months, and UK Customer (a dummy variable that indicates whether the customer comes from the UK). Since our objective is to predict whether a customer will buy products or not in the last three months, it seems that it is important to know the elapsed number of months since the last purchase. In addition, we observe that the customer origin plays an important role; customers in the UK tend to buy less than foreign customers. Finally, similar conclusions to the ones shown in the rest of the examples can be stated with respect to the sensitivity analysis.

In this example, it is even more clear that the choice of the parameter C is a crucial issue for obtaining good accuracy. See Fig. 11c in Appendix A for more details.

Figures 11b and c show again that the elimination threshold δ and the number of iterations do not affect the effectiveness of our approach. In Fig. 12 we can observe the evolution of the values of the different bandwidths which converge in a small number of iterations.

4 Conclusions and extensions

In this paper, we have shown how the well-known SVM technique can be embedded with a feature selection strategy to get the most informative covariates of hybrid functional data. In fact, we have compared our approach with 17 benchmark methodologies from the literature, and our proposal achieves the best average accuracy. In our proposed approach, we have modified the standard Gaussian kernel by associating a bandwidth with each variable. Such bandwidths and the rest of the SVM parameters are sought via a bilevel optimization problem solved with an alternating approach. Instead of minimizing the misclassification rate, we propose maximizing the Pearson correlation between the class label and the score. Other measures such as the correlation in [82] can also be applied. Our methodology can also be used if all the components of the data are functions, i.e. the pure multivariate functional data case. A sensitivity analysis of the setting parameters involved in our approach was made to show its robustness. We observe that the choice of the parameter C is critical to yielding good classification rates. Some standard cross-validation methods may be used to get a good value of C. In contrast, the elimination threshold and the maximum number of iterations allowed in the alternating approach do not affect the accuracy obtained. Moreover, the values of the bandwidths associated with the features converge in few iterations to their final value. We have restricted ourselves to the binary classification problem. The extension to other related fields, such as multiclass classification or regression, [10], deserves further study. In our proposal, we use standard optimization techniques solve Problems (1) and (6). As a future research line, we can develop more efficient optimization strategies compatible with the world of Big Data, e.g. methodologies applied to Problem (1) which do not need the computation of the whole kernel matrix, or the use of stochastic gradients to iterate in the bandwidth parameters of Problem (6). Finally, the application of our approach to other real-world contexts, such as the field of medicine, should be analyzed too.

Acknowledgements Research partially supported by research grants MTM2015-65915-R (Ministerio de Ciencia e Innovación, Spain), P11-FQM-7603, P18-FR-2369, FQM329 (Junta de Andalucía, Spain), FPU (Ministerio de Educación, Cultura y Deporte), VI PPITUS (Universidad de Sevilla), all with EU ERDF funds, as well as FBBVA-COSECLA. Moreover, thank the team of the Scientific Computing Center of Andalucía (CICA) for the computing services provided. This support is gratefully acknowledged by the first author. The second author would like to thank ANID, FONDECYT project 1200221, and the Complex Engineering Systems Institute (ANID, PIA, FB0816).

Compliance with Ethical Standards

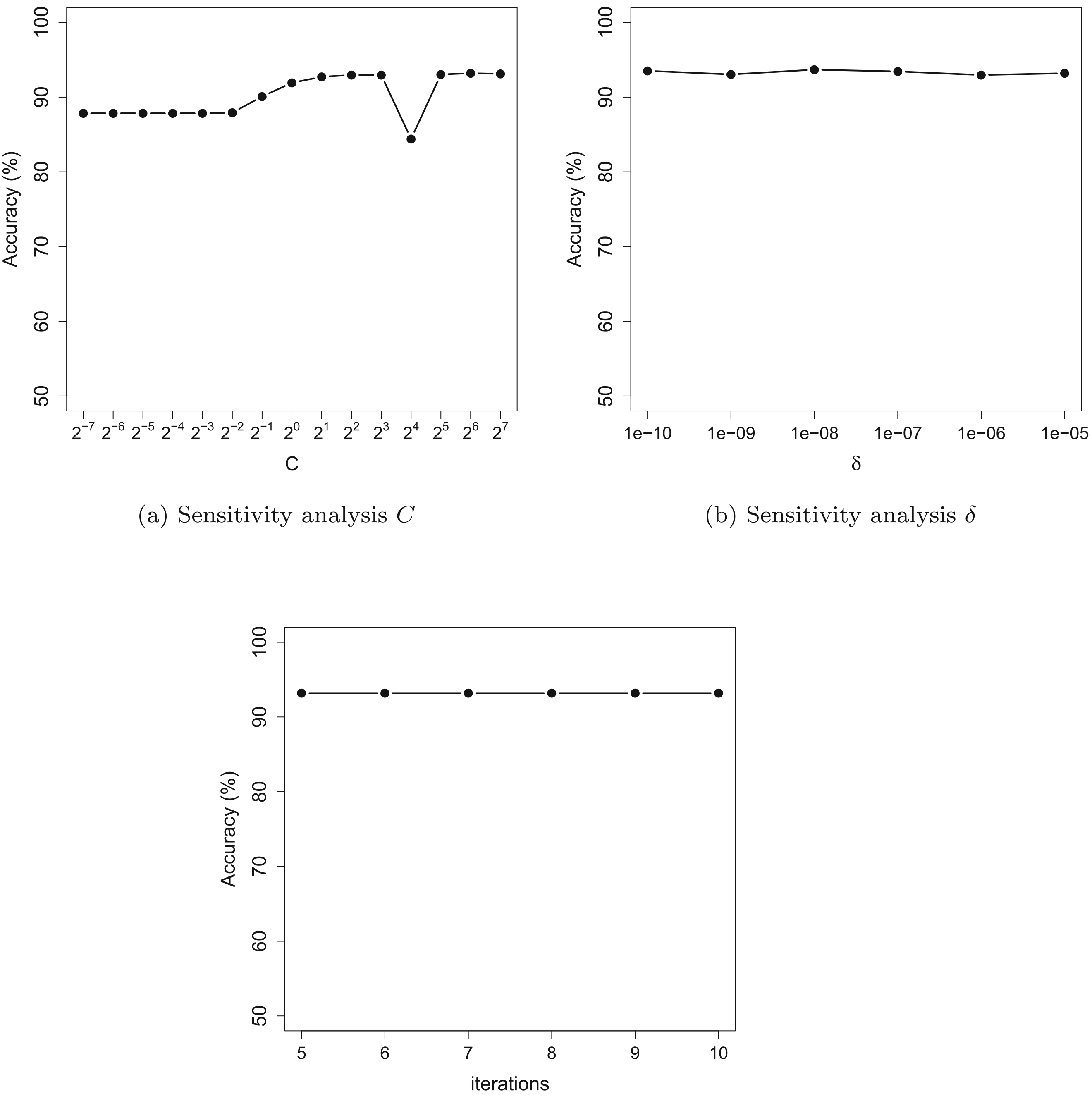
Conflict of interests The authors declare that they have no conflict of interest.

Appendix A: Sensitivity analysis

In order to study the robustness of our proposed algorithm with respect to the parameters involved, we ran a sensitivity analysis. We tested how sensitive our methodology is to the regularization parameter C, the threshold at which the features are removed δ, the maximum number of iterations of the alternating approach, and the bandwidths ω_v, $v = 1, \ldots, p + q$. First, we ran five times the alternating approach of Algorithm 1 to test the sensitivity of the algorithm with respect to the parameter C, computing the average accuracy on s_3. Second, the sensitivity analysis for the elimination threshold δ is performed by running Algorithm 1 five times for the values given in the set $\{10^{-10}, \ldots, 10^{-5}\}$ in logarithmic scale. The average accuracy is estimated on s_3. Third, the maximum number of iterations of the alternating approach may affect the classification rates. In order to check the robustness of our proposal, Algorithm 1 is run five times with the maximum number of iterations belonging to the set $\{5, \ldots, 10\}$. The average accuracy measured on the sample s_3 is then computed. Finally, we studied the convergence of the bandwidths. Note that in this paper, convergence does not mean that the bandwidths tend to the same value in all the runs, but that they are greater or less than δ, and yield the same features in most of the cases. For each of the five times that Algorithm 1 was run, the optimal values of the bandwidths after the alternating approach were obtained. The goal is to assess the importance of the variables visually. In all the sensitivity analysis studied, the remaining parameters which were not under study took the values given in Section 3.2. For instance, when the sensitivity with respect to C was analyzed, the elimination threshold was equal to 10^{-5}, and the maximum number of iterations of the alternating approach was set to five. Plots of results of the sensitivity analysis for all the parameters above mentioned in the *batch* data set are depicted in Figs. 5 and 6. Figures 7 and 8 depict the results for *trigonometric* data set, whereas the results of the *pen* data set are shown in Figs. 9 and 10. Finally, Figs. 11 and 12 show the sensitivity analysis of the *retail* data set.

Appendix B: Analysis of sensitivity, specificity and area under the curve

This section provides three tables with new performance metrics, namely sensitivity (Table 5), specificity (Table 6) and Area under the Curve (Table 7). More details about the conclusions derived from these tables can be seen in Section 3.4.

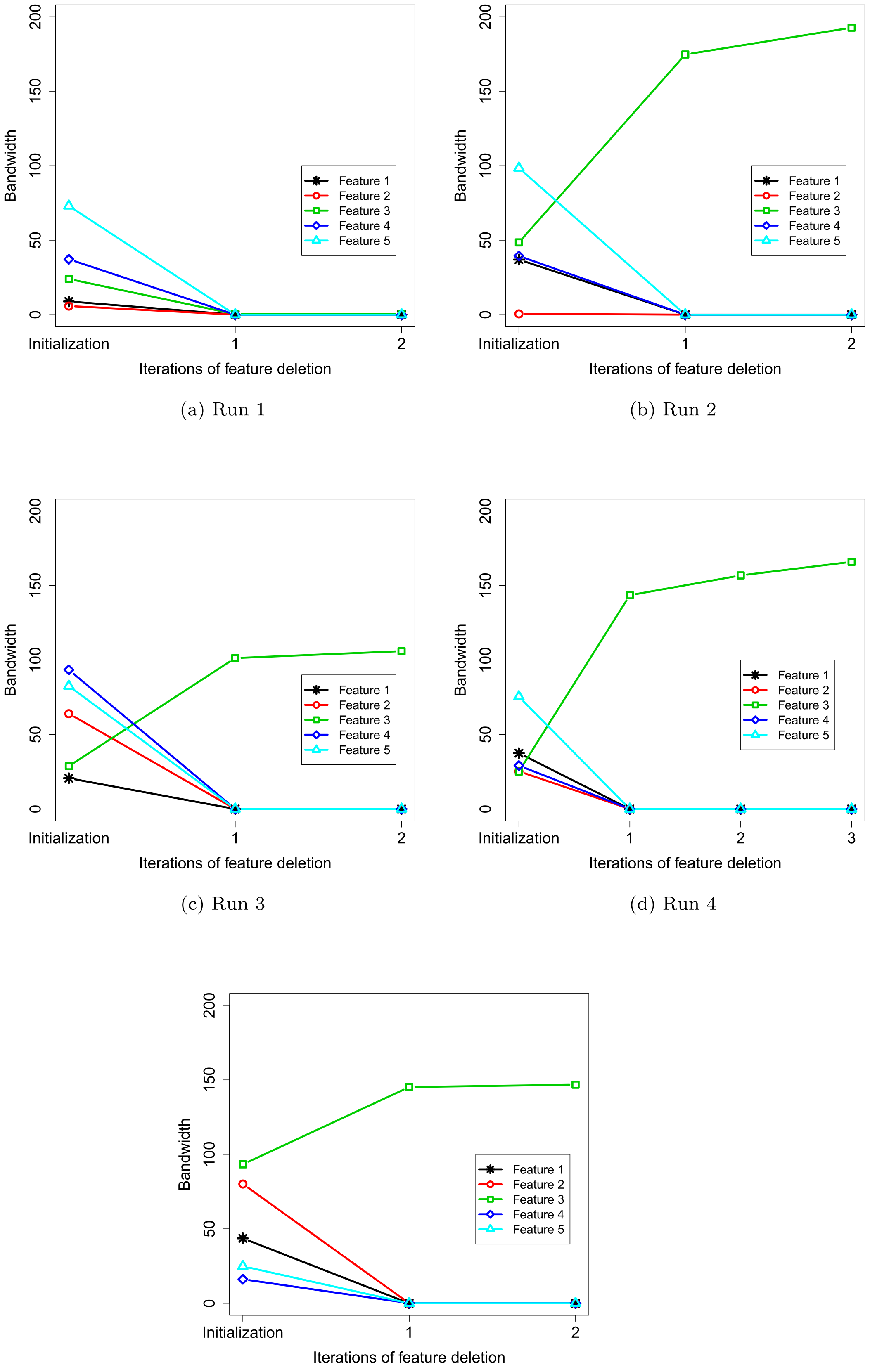


(a) Sensitivity analysis C

(b) Sensitivity analysis δ

(c) Sensitivity analysis number iterations

Fig. 5 Results of the sensitivity analysis for the *batch* data set



(a) Run 1

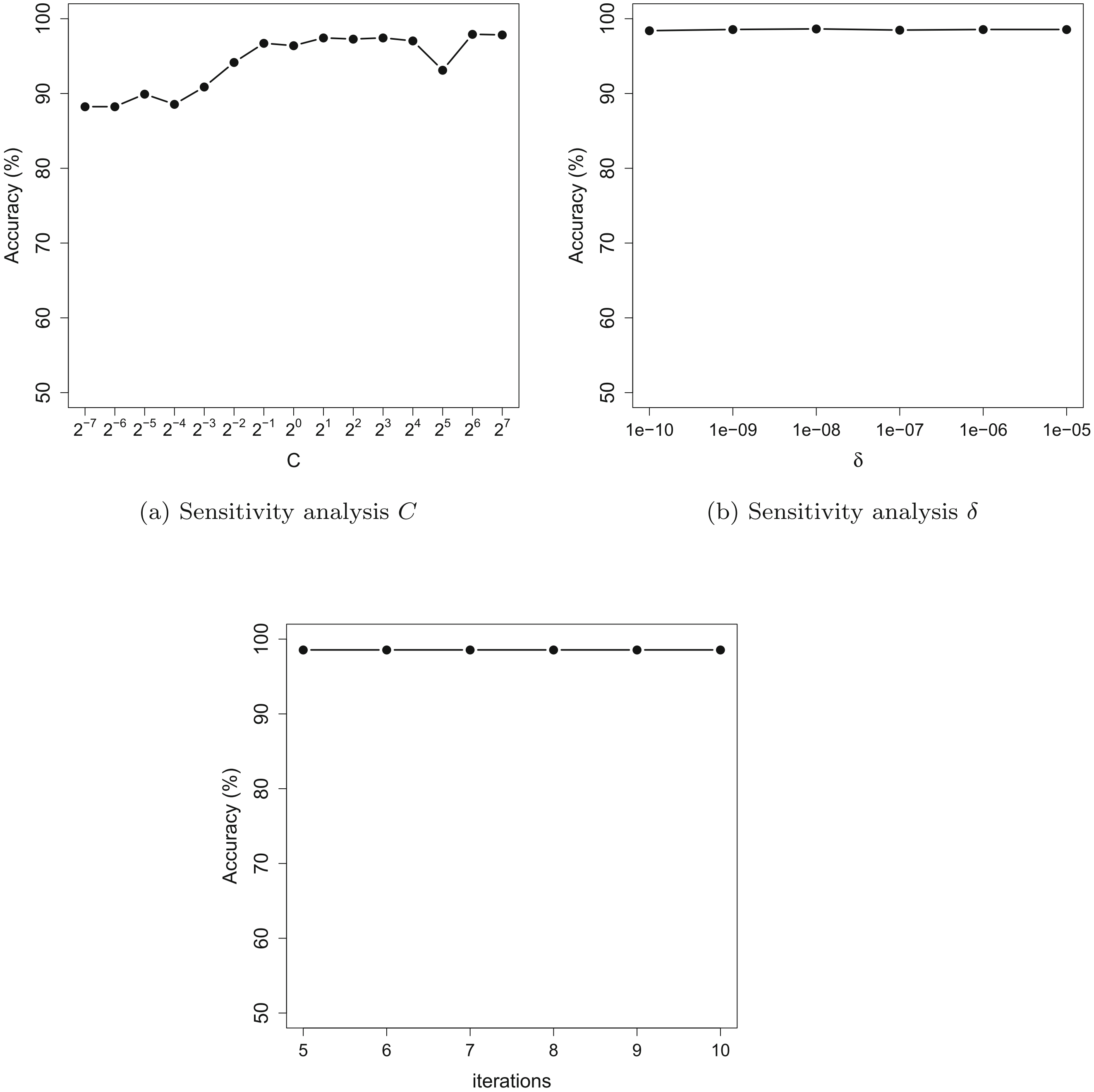
(b) Run 2

(c) Run 3

(d) Run 4

(e) Run 5

Fig. 6 Convergence of the bandwidths for the *batch* data set

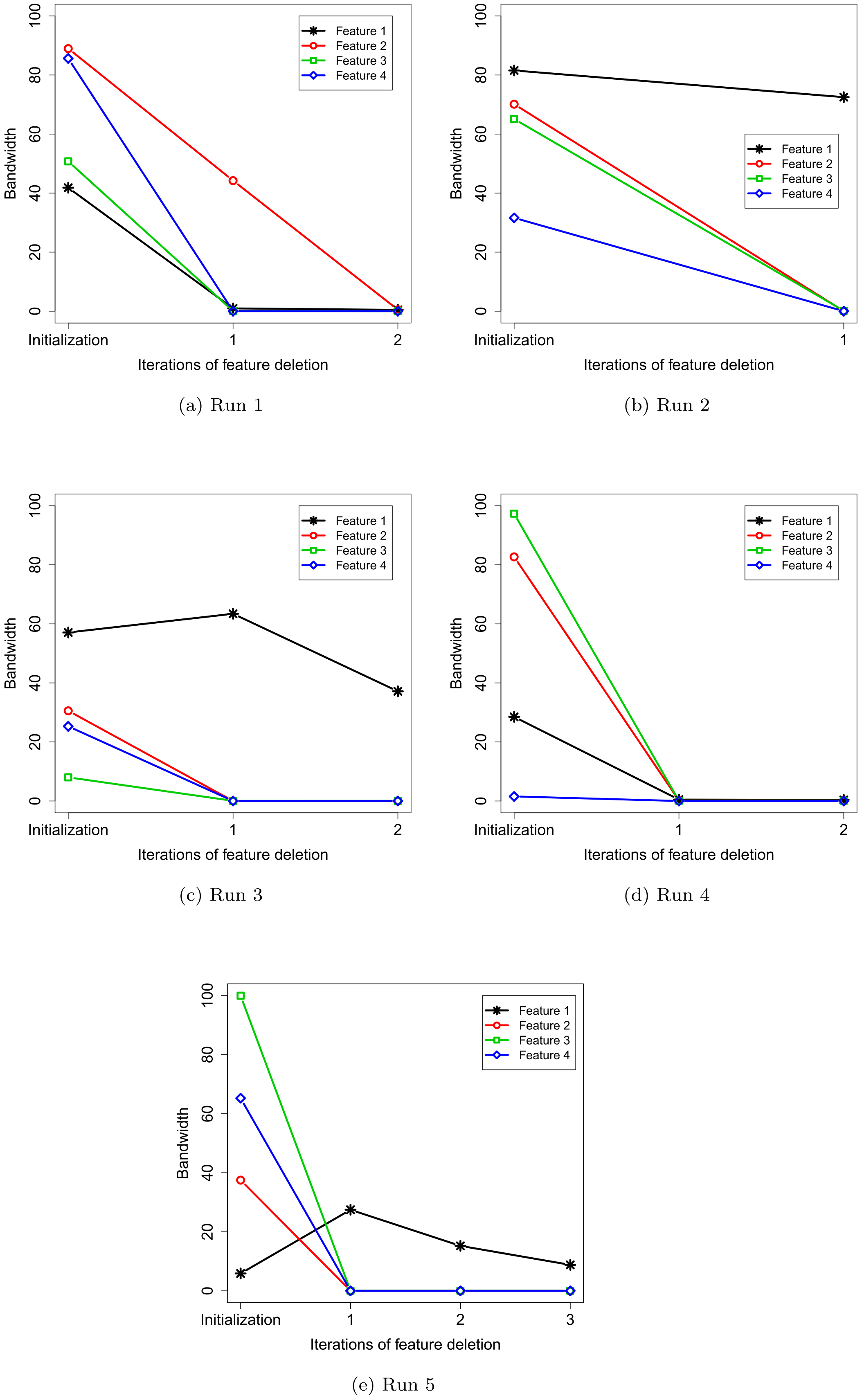


(a) Sensitivity analysis C

(b) Sensitivity analysis δ

(c) Sensitivity analysis number iterations

Fig. 7 Results of the sensitivity analysis for the *trigonometric* data set



(a) Run 1

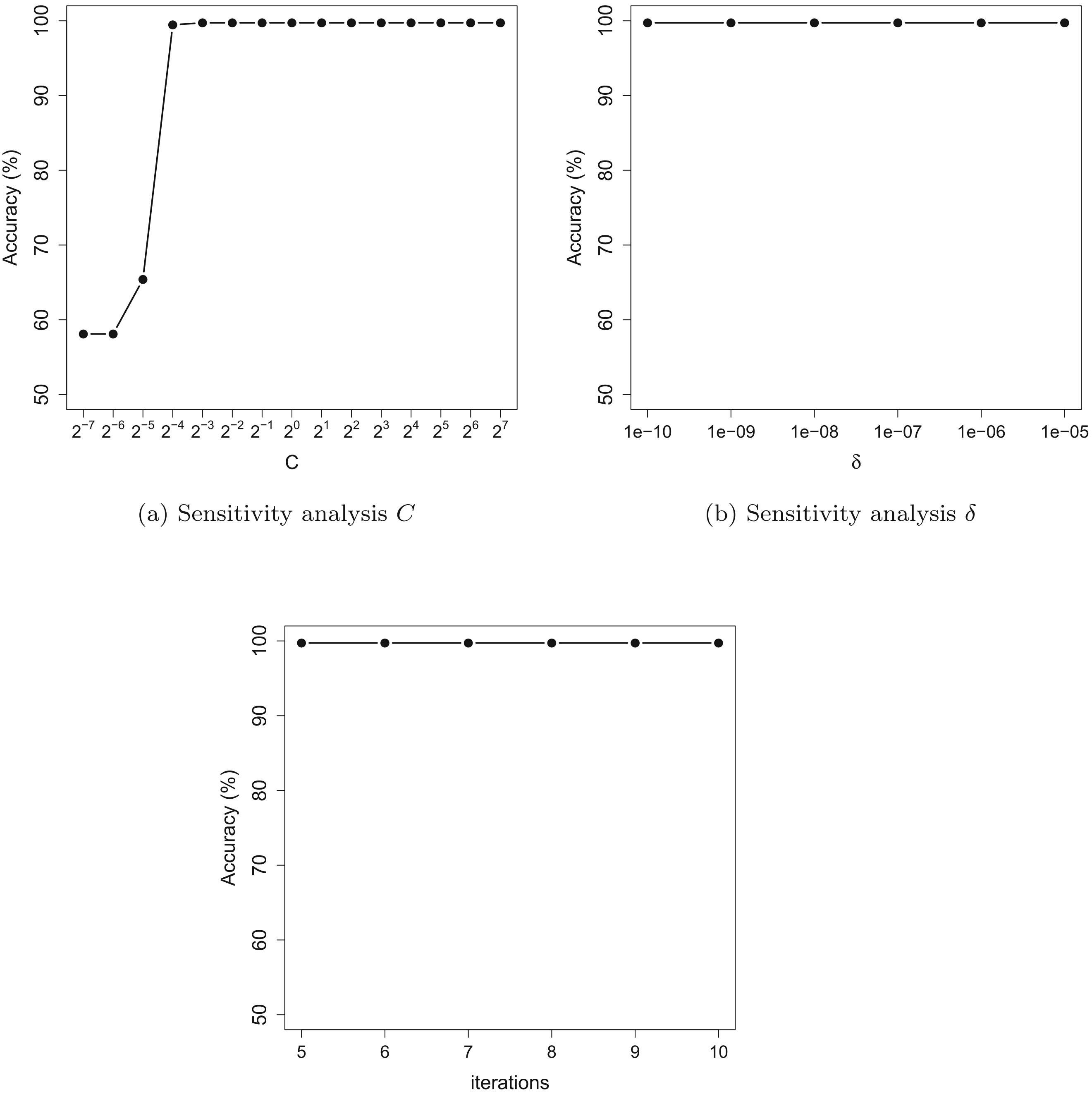
(b) Run 2

(c) Run 3

(d) Run 4

(e) Run 5

Fig. 8 Convergence of the bandwidths for the *trigonometric* data set

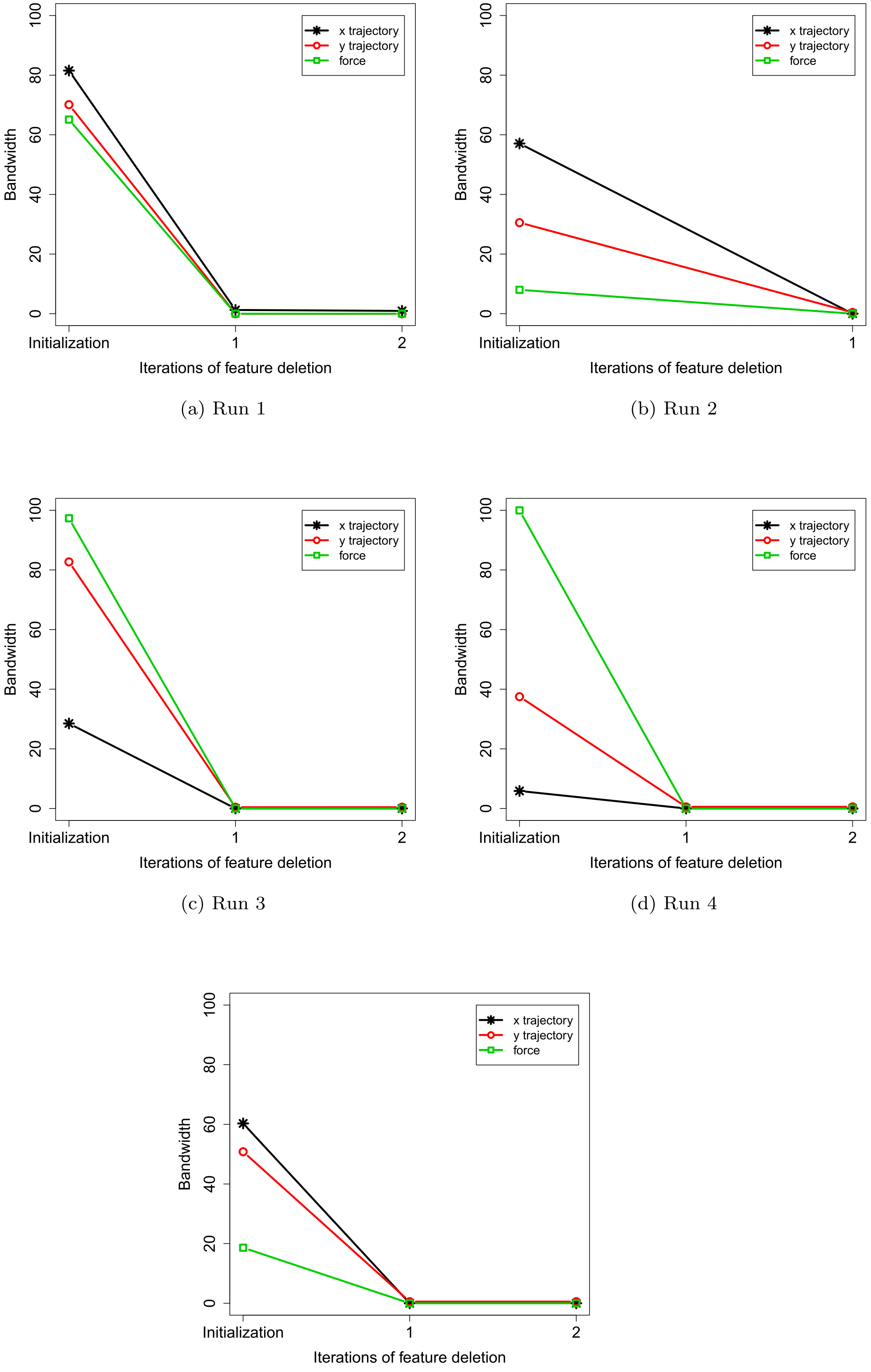


(a) Sensitivity analysis C

(b) Sensitivity analysis δ

(c) Sensitivity analysis number iterations

Fig. 9 Results of the sensitivity analysis for the *pen* data set



(a) Run 1

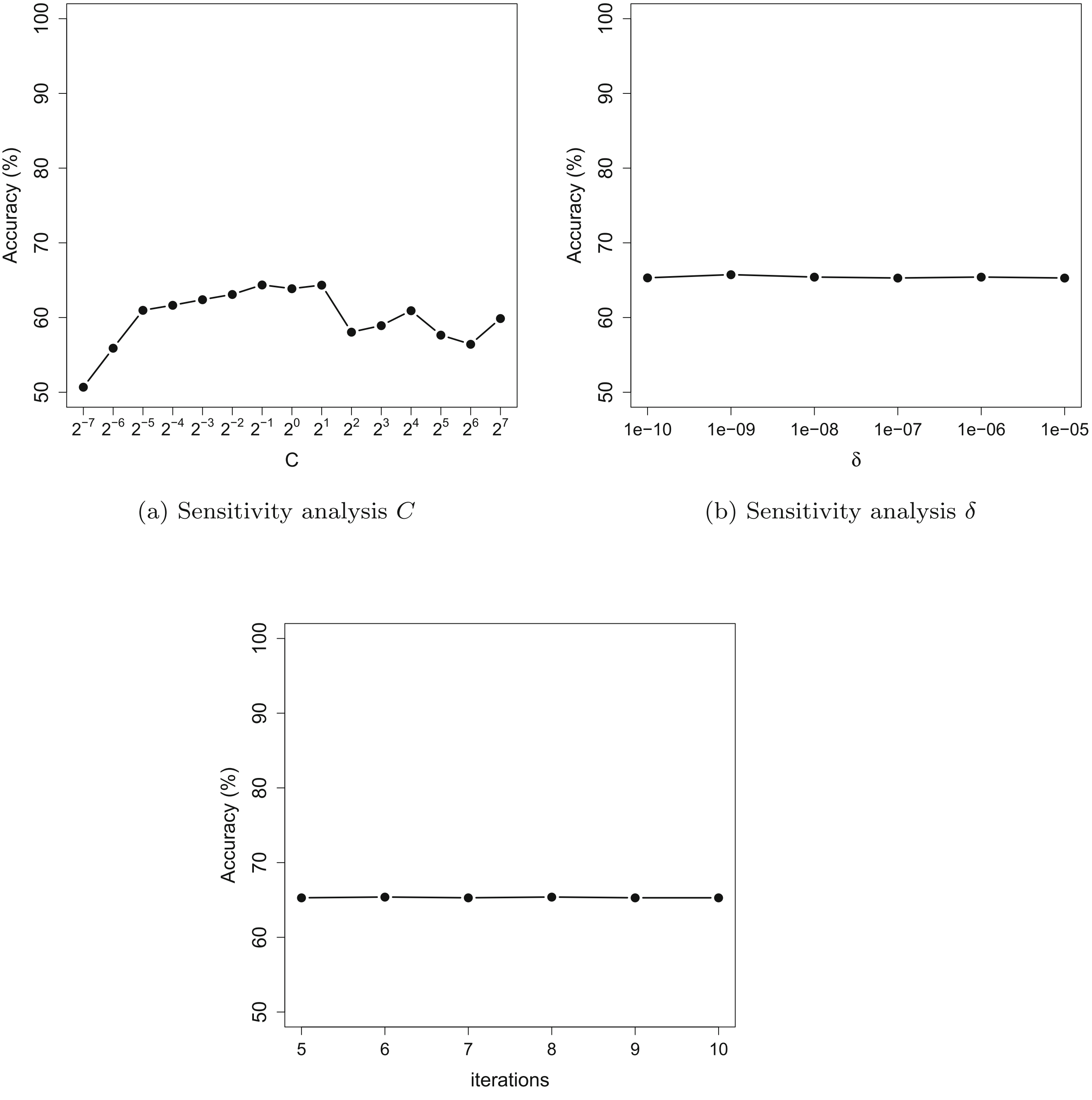
(b) Run 2

(c) Run 3

(d) Run 4

(e) Run 5

Fig. 10 Convergence of the bandwidths for the *pen* data set

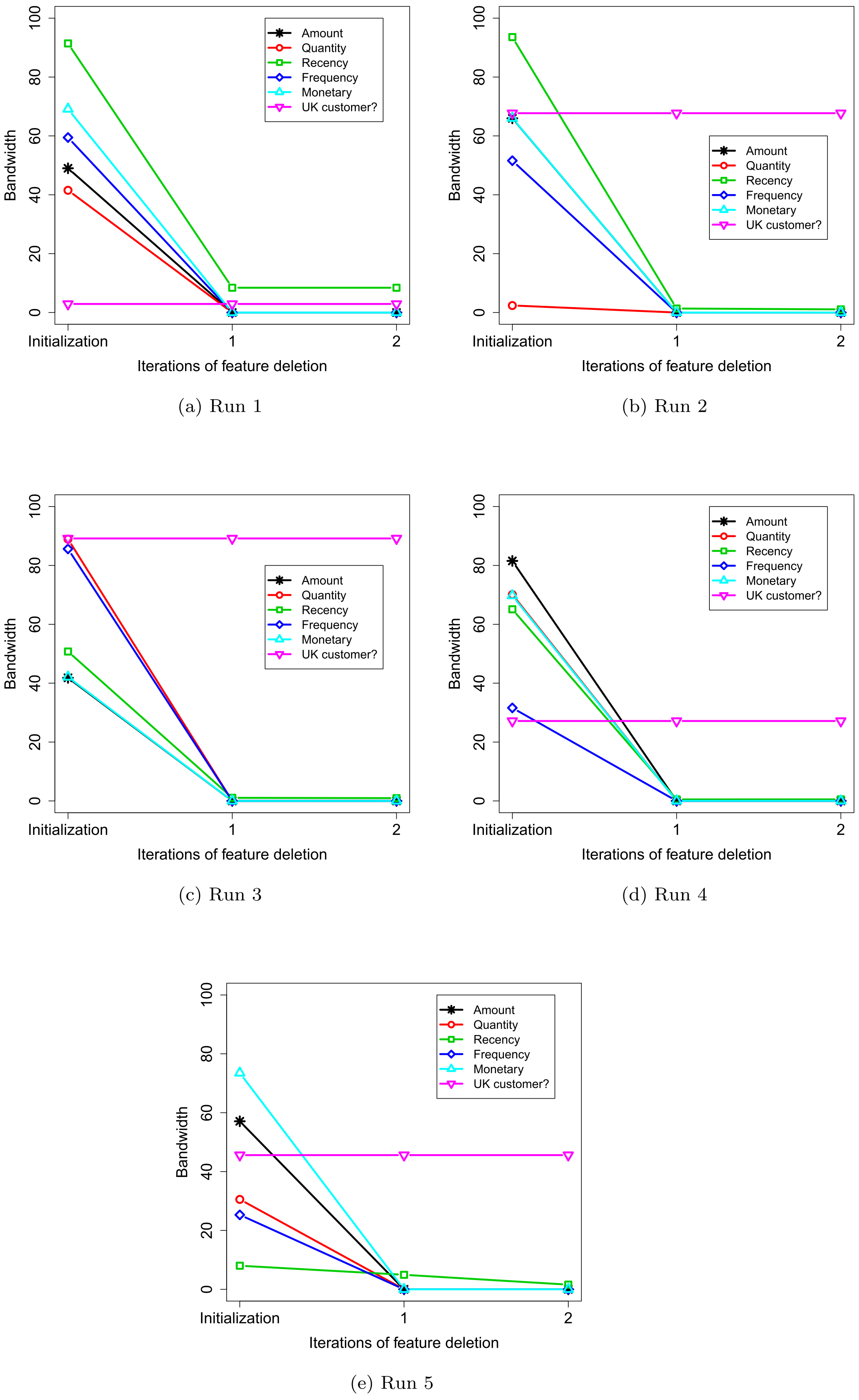


(a) Sensitivity analysis C

(b) Sensitivity analysis δ

(c) Sensitivity analysis number iterations

Fig. 11 Results of the sensitivity analysis for the *retail* data set



(a) Run 1

(b) Run 2

(c) Run 3

(d) Run 4

(e) Run 5

Fig. 12 Convergence of the bandwidths for the *retail* data set

Table 5 Result summary

Data set	batch	trigonometric	pen	retail
Alt. appr.	**0.96**	**0.98**	**1**	**0.64**
FSVM	0.75	0.94	**1**	0.57
ℓ_2-SVM (4 dim)	0.75	0.81	0.98	0.57
ℓ_2-SVM (disc)	0.88	0.97	**1**	0.59
ℓ_2-LR$_p$	0.95	0.88	**1**	**0.64**
$\ell_2\ell_2$-SVM$_d$	0.93	0.84	**1**	0.61
$\ell_2\ell_2$-SVM$_p$	0.94	0.88	**1**	**0.64**
$\ell_2\ell_1$-SVM$_d$	0.92	0.88	**1**	0.62
SVM$_{CS}$	0.93	0.80	**1**	0.63
$\ell_1\ell_2$-SVM	0.93	0.92	**1**	**0.64**
ℓ_1-LR	0.93	0.91	**1**	**0.64**
ℓ_2-LR$_d$	0.93	0.90	**1**	**0.64**
$\mathcal{X}^2$ test	0.54	**0.98**	**1**	0.61
information_gain	0.54	**0.98**	**1**	0.61
kruskal_test	0.55	**0.98**	**1**	0.63
min_depth	0.51	**0.98**	**1**	0.62
rf_importance	0.52	**0.98**	**1**	0.62
variance	0.53	0.97	**1**	0.61

Sensitivity as performance metric

Table 6 Result summary

Data set	batch	trigonometric	pen	retail
Alt. appr.	**0.95**	**0.99**	0.98	0.66
FSVM	0.81	0.98	0.98	0.79
ℓ_2-SVM (4 dim)	0.78	0.88	**0.99**	0.77
ℓ_2-SVM (disc)	0.92	0.94	0.98	**0.78**
ℓ_2-LR$_p$	0.94	0.94	0.98	0.69
$\ell_2\ell_2$-SVM$_d$	0.92	0.92	0.98	0.60
$\ell_2\ell_2$-SVM$_p$	0.93	0.95	0.98	0.69
$\ell_2\ell_1$-SVM$_d$	0.92	0.95	0.98	0.66
SVM$_{CS}$	0.92	0.81	0.98	0.72
$\ell_1\ell_2$-SVM	0.93	0.93	0.98	0.70
ℓ_1-LR	0.93	0.93	0.98	0.69
ℓ_2-LR$_d$	0.93	0.93	0.98	0.70
$\mathcal{X}^2$ test	0.54	0.98	0.98	0.65
information_gain	0.54	0.98	0.98	0.65
kruskal_test	0.55	0.98	0.98	0.65
min_depth	0.51	0.98	0.98	0.65
rf_importance	0.52	0.97	0.97	0.66
variance	0.53	0.98	0.98	0.66

Specificity as performance metric

Table 7 Result summary

Data set	batch	trigonometric	pen	retail
Alt. appr.	**0.95**	**0.99**	**0.99**	0.64
FSVM	0.74	0.96	**0.99**	0.63
ℓ_2-SVM (4 dim)	0.76	0.83	0.98	0.62
ℓ_2-SVM (disc)	0.90	0.96	0.98	0.64
ℓ_2-LR$_p$	0.95	0.90	**0.99**	0.66
$\ell_2\ell_2$-SVM$_d$	0.92	0.88	**0.99**	0.60
$\ell_2\ell_2$-SVM$_p$	0.93	0.91	**0.99**	**0.67**
$\ell_2\ell_1$-SVM$_d$	0.92	0.91	**0.99**	0.64
SVM$_{CS}$	0.92	0.80	**0.99**	0.66
$\ell_1\ell_2$-SVM	0.93	0.92	**0.99**	**0.67**
ℓ_1-LR	0.93	0.92	**0.99**	**0.67**
ℓ_2-LR$_d$	0.93	0.92	**0.99**	**0.67**
$\mathcal{X}^2$ test	0.54	**0.99**	**0.99**	0.63
information_gain	0.54	**0.99**	0.98	0.63
kruskal_test	0.55	**0.99**	0.98	0.64
min_depth	0.51	0.98	**0.99**	0.63
rf_importance	0.53	0.98	0.98	0.63
variance	0.53	0.98	**0.99**	0.63

Area Under the Curve (AUC) as performance metric

References

1. Alber M, Zimmert J, Dogan U, Kloft M (2017) Distributed optimization of multi-class svms. Plos One 12(6):1–18
2. Baesens B (2014) Analytics in a Big Data World. Wiley
3. Baíllo A, Cuevas A, Fraiman R (2011) Classification methods for functional data
4. Berrendero J, Justel A, Svarc M (2011) Principal components for multivariate functional data. Comput Stat Data An 55(9):2619–2634
5. Berrendero JR, Cuevas A, Torrecilla JL (2016) Variable selection in functional data classification: a maxima-hunting proposal. Stat Sin 26:619–638
6. Bischl B, Lang M, Kotthoff L, Schiffner J, Richter J, Studerus E, Casalicchio G, Jones ZM (2016) mlr: Machine learning in. R. J Mach Learn Res 17(170):1–5
7. Blanquero R, Carrizosa E, Chis O, Esteban N, Jiménez-Cordero A, Rodríguez JF, Sillero-Denamiel MR (2016) On extreme concentrations in chemical reaction networks with incomplete measurements. Ind Eng Chem Res 55:11417–11430
8. Blanquero R, Carrizosa E, Jiménez-Cordero A, Rodríguez JF (2016) A global optimization method for model selection in chemical reactions networks. Comput Chem Eng 93:52–62
9. Blanquero R, Carrizosa E, Jiménez-Cordero A, Martín-Barragán B (2019) Functional-bandwidth kernel for Support Vector Machine with functional data: an alternating optimization algorithm. European J Op Res 275:195–207
10. Blanquero R, Carrizosa E, Jiménez-Cordero A, Martín-Barragán B (2019) Selection of time instants and intervals with support

vector regression for multivariate functional data. Tech. rep., University of Seville - University of Málaga - University of Edinburgh, available at https://www.researchgate.net/publication/327552293_Selection_of_Time_Instants_and_Intervals_with_Support_Vector_Regression_for_Multivariate_Functional_Data

11. Blanquero R, Carrizosa E, Jiménez-Cordero A, Martín-Barragán B (2019) Variable selection in classification for multivariate functional data. Inform Sci 481:445–462
12. Boente G, Fraiman R (2000) Kernel-based functional principal components. Stat Probab Lett 48(4):335–345
13. Bommert A, Sun X, Bischl B, Rahnenführer J, Lang M (2020) Benchmark for filter methods for feature selection in high-dimensional classification data. Comput Stat Data Anal 106839:143
14. Bradley P, Mangasarian O (1998) Feature selection via concave minimization and support vector machines. In: Machine Learning proceedings of the fifteenth International Conference (ICML'98). San Francisco, California, Morgan Kaufmann, pp 82–90
15. Bugeau A, Pérez P (2007) Bandwidth selection for kernel estimation in mixed multi-dimensional spaces. Tech. rep., INRIA, available at https://arxiv.org/abs/0709.1920v2
16. Cai J, Luo J, Wang S, Yang S (2018) Feature selection in machine learning: a new perspective. Neurocomputing 300:70–79
17. Cai TT, Hall P (2006) Prediction in functional linear regression. Annals Stat 34(5):2159–2179
18. Carrizosa E, Martín-Barragán B, Romero-Morales D (2014) A nestedheuristic for parameter tuning in support vector machines. Comput Ops Res 43:328–334
19. Cauwenberghs G, Poggio T (2001) Incremental and decrementalsupport vector machine learning. In: Advances in neural information processing systems, pp 409–415
20. Chen D, Sain SL, Guo K (2012) Data mining for the online retail industry: a case study of RFM model-based customer segmentation using data mining. J Database Mark Cust Strateg Manag 19(3):197–208
21. Chen Q, Wynne R, Goulding P, Sandoz D (2000) The application of principal component analysis and kernel density estimation to enhance process monitoring. Control Eng Pract 8(5):531–543
22. Chiou JM, Chen YT, Yang YF (2014) Multivariate functional principal component analysis: a normalization approach. Stat Sin 24(4):1571–1596
23. Colson B, Marcotte P, Savard G (2007) An overview of bilevel optimization. Ann Oper Res 153(1):235–256
24. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297
25. Cristianini N, Shawe-Taylor J (2000) An introduction to Support Vector Machines and other kernel-based learning methods. Cambridge University Press
26. Cuesta-Albertos JA, Fraiman R (2007) Impartial trimmed k-means for functional data. Comput Stat Data An 51(10):4864–4877
27. Cuevas A, Febrero M, Fraiman R (2002) Linear functional regression: the case of fixed design and functional response. Can J Stat 30(2):285–300
28. Delaigle A, Hall P (2012) Achieving near perfect classification for functional data. J R Stat Soc: Series B Stat Methodol 74(2):267–286
29. Demšar J (2006) Statisticalcomparisons of classifiers over multiple data sets. J Mach Learn Res 7:1–30
30. Dheeru D, Karra-Taniskidou E (2017) UCI machine learning repository http://archive.ics.uci.edu/ml
31. Duan KB, Rajapakse JC, Wang H, Azuaje F (2005) Multiple svm-rfe for gene selection in cancer classification with expression data. IEEE Trans NanoBioscience 4(3):228–234
32. Duda R (2001) Pattern Classification. Wiley-Interscience Publication, Stork D
33. Duong T, Cowling A, Koch I, Wand M (2008) Feature significance for multivariate kernel density estimation. Comput Stat Data An 52(9):4225–4242
34. Fan RE, Chang KW, Hsieh CJ, Wang XR, Lin CJ (2008) LIBLINEAR: A library for large linear classification. J Mach Learn Res 9:1871–1874
35. Febrero-Bande M, González-Manteiga W, de la Fuente MO (2017) Variable selection in functional additive regression models. In: Aneiros G, G Bongiorno E, Cao R, Vieu P (eds) Functional statistics and related fields. Springer International Publishing, Cham, pp 113-122
36. Ferraty F, Vieu P (2006) Nonparametric functional data analysis: theory and practice
37. García S, Fernández A, Luengo J, Herrera F (2010) Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. Information Sciences 180(10):2044–2064. special Issue on Intelligent Distributed Information Systems
38. Gaur P, Pachori RB, Wang H, Prasad G (2018) A multi-class EEG-based BCI classification using multivariate empirical mode decomposition based filtering and Riemannian geometry. Expert Syst Appl 95:201–211
39. Gómez-Verdejo V, Verleysen M, Fleury J (2007) Information-theoreticfeature selection for functional data classification. Neurocomputing Financial Engineering Computational and Ambient Intelligence IWANN 72(16):3580–3589
40. Gregorutti B, Michel B, Saint-Pierre P (2017) Correlation and variable importance in random forests. Stat Comput 27(3):659–678
41. Guyon I, Weston J, Barnhill S, Vapnik V (2002) Gene selection for cancer classification using Support Vector Machines. Mach Learn 46(1-3):389–422
42. Guyon I, Gunn S, Nikravesh M, Zadeh LA (2006) Feature extraction foundations and applications. Springer, Berlin
43. Hajewski J, Oliveira S, Stewart D (2018) Smoothed hinge loss and ?1 support vector machines. In: 2018 IEEE International Conference on Data Mining Workshops ICDMW, pp 1217–1223
44. Hall P, Hosseini-Nasab M (2006) On properties of functional principal components analysis. J R Stat Soc: Series B Stat Methodol 68(1):109–126
45. Hancer E, Xue B, Zhang M (2018) Differential evolution for filter feature selection based on information theory and feature ranking. Knowl-Based Syst 140:103–119
46. Happ C, Greven S (2018) Multivariate functional principal component analysis for data observed on different dimensional domains. J Am Stat Assoc 113(522):649–659
47. Hubert M, Rousseeuw PJ, Segaert P (2015) Multivariate functional outlier detection. Stat Methods Appl 24(2):177–202
48. Hubert M, Rousseeuw P, Segaert P (2017) Multivariate and functional classification using depth and distance. ADAC 11(3):445–466
49. Jacques J, Preda C (2014) Model-based clustering for multivariate functional data. Comput Stat Data An 71:92–106
50. James GM, Hastie TJ (2001) Functional linear discriminant analysis for irregularly sampled curves. J R Stat Soc: Series B Stat Methodol 63(3):533–550
51. Kadri H, Duflos E, Preux P, Canu S, Davy M (2010) Nonlinearfunctional regression: a functional RKHS approach. In: International Conference on Artificial Intelligence and Statistics, pp 374–380
52. Kayano M, Dozono K, Konishi S (2010) Functional cluster analysis via orthonormalized gaussian basis expansions and its application. J Classif 27(2):211–230
53. Ke W, Wu C, Wu Y, Xiong NN (2018) A new filter feature selection based on criteria fusion for gene microarray data. IEEE Access 6:61065–61076

54. Keerthi SS, Lin CJ (2003) Asymptotic behaviors of support vector machines with gaussian kernel. Neural Comput 15(7):1667–1689
55. Labani M, Moradi P, Ahmadizar F, Jalili M (2018) A novel multivariate filter method for feature selection in text classification problems. Eng Appl Artif Intell 70:25–37
56. Li B, Yu Q (2008) Classification of functional data: a segmentation approach. Comput Stat Data An 52(10):4790–4800
57. Li PL, Chiou JM (2011) Identifying cluster number for subspace projected functional data clustering. Comput Stat Data An 55(6):2090–2103
58. Li W, Lederer J (2019) Tuning parameter calibration for ℓ_1-regularized logistic regression. J Stat Plan Infer 202:80–98
59. López J, Maldonado S (2018) Robust twin support vector regression via second-order cone programming. Knowl-Based Syst 152:83–93
60. Mafarja M, Mirjalili S (2018) Whale optimization approaches for wrapper feature selection. Appl Soft Comput 62:441–453
61. Maldonado S, López J (2017) Synchronized feature selection for support vector machines with twin hyperplanes. Knowl-Based Syst 132:119–128
62. Maldonado S, Weber R, Basak J (2011) Simultaneous feature selection and classification using kernel-penalized support vector machines. Inf Sci 181(1):115–128
63. Maldonado S, Carrizosa E, Weber R (2015) Kernel penalized k-means: a feature selection method based on kernel k-means. Inf Sci 322:150–160
64. Maldonado S, Merigó J, Miranda J (2018) Redefining support vector machines with the ordered weighted average. Knowl-Based Syst 148:41–46
65. Martín-Barragán B, Lillo R, Romo J (2014) Interpretable support vector machines for functional data. Eur J Oper Res 232(1):146–155
66. Meng Y, Liang J, Qian Y (2016) Comparison study of orthonormal representations of functional data in classification. Knowl-Based Syst 97:224–236
67. Muñoz A, González J (2010) Representing functional data using support vector machines. Pattern Recogn Lett 31(6):511–516
68. Muthusankar D, Kalaavathi B, Kaladevi P (2019) High performance feature selection algorithms using filter method for cloud-based recommendation system. Clust Comput 22(1):311–322
69. Pecha M, Horák D (2020) Analyzing ℓ_1-loss and ℓ_2-loss support vector machines implemented in PERMON toolbox. In: Zelinka I, Brandstetter P, Trong Dao T, Hoang Duy V, Kim SB (eds) Recent advances in electrical engineering and related sciences: theory and application, vol 2018. Springer International Publishing, Cham, pp 13–23
70. Preda C, Saporta G, Lévéder C (2007) PLS Classification of functional data. Comput Stat 22(2):223–235
71. Ramsay JO, Silverman BW (2002) Applied functional data analysis: methods and case studies Springer Series in Statistics, vol 77. Springer-Verlag
72. Ramsay JO, Silverman BW (2005) Functional data analysis, 2nd edn. Springer Series in Statistics, Springer-Verlag
73. Ratcliffe SJ, Heller GZ, Leader LR (2002) Functional data analysis with application to periodically stimulated foetal heart rate data. ii: Functional logistic regression. Stat Med 21(8):1115–1127
74. Rossi F, Villa N (2006) Support vector machine for functional data classification. Neurocomputing 69(7):730–742
75. Rossi F, Villa N (2008) Recent advances in the use of SVM for functional data classification. Physica-Verlag HD, Heidelberg, pp 273-280
76. Sain SR (2002) Multivariate locally adaptive density estimation. Comput Stat Data An 39(2):165–186
77. Salaheldin R, El Gayar N (2011) Multiple classifiers for time series classification using adaptive fusion of feature and distance based methods UKCI, vol 2011, p 114
78. Strle B, Mozina M, Bratko I (2009) Qualitative approximation to dynamic time warping similarity between time series data. In: Proceedings of the Workshop on Qualitative Reasoning
79. Core Team R (2017) R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria https://www.R-project.org/
80. Temel T (2017) A new classification algorithm: optimally generalized learning vector quantization (oglvq). Neural Network World 27(6):569–576
81. Tokushige S, Yadohisa H, Inada K (2007) Crisp and fuzzy k-means clustering algorithms for multivariate functional data. Comput Stat 22(1):1–16
82. Torrecilla Noguerales JL (2015) On the theory and practice of variable selection for functional data PhD thesis Universidad Autónoma de Madrid
83. Tubishat M, Abushariah MAM, Idris N, Aljarah I (2019) Improved whale optimization algorithm for feature selection in arabic sentiment analysis. Appl Intell 49(5):1688–1707
84. Vapnik V (1998) Statistical Learning Theory. Wiley
85. Wang H, Yao M (2015) Fault detection of batch processes based on multivariate functional kernel principal component analysis. Chemometr Intell Lab Syst 149:78–89
86. Zou F, Wang Y, Yang Y, Zhou K, Chen Y, Song J (2015) Supervised feature learning via ℓ_2-norm regularized logistic regression for 3D object recognition. Neurocomputing 151:603–611

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.