



Combining unsupervised and supervised learning techniques for enhancing the performance of functional data classifiers

Fabrizio Maturo¹ · Rosanna Verde¹

Received: 21 April 2022 / Accepted: 21 June 2022 / Published online: 25 July 2022
© The Author(s) 2022

Abstract

This paper offers a supervised classification strategy that combines functional data analysis with unsupervised and supervised classification methods. Specifically, a two-steps classification technique for high-dimensional time series treated as functional data is suggested. The first stage is based on extracting additional knowledge from the data using unsupervised classification employing suitable metrics. The second phase applies functional supervised classification of the new patterns learned via appropriate basis representations. The experiments on ECG data and comparison with the classical approaches show the effectiveness of the proposed technique and exciting refinement in terms of accuracy. A simulation study with six scenarios is also offered to demonstrate the efficacy of the suggested strategy. The results reveal that this line of investigation is compelling and worthy of further development.

Keywords Functional data analysis · Functional supervised classification · Functional k-means · Functional random forest · Augmented labels

1 Introduction

Learning from high-dimensional data is a hot topic in supervised and unsupervised classification frameworks. There are various reasons why the statistical literature is rapidly evolving on the subject. Technological progress increasingly stimulates research and the production of devices to gather data in large quantities in many human activities and fields of study, such as the medical, ecological, telephone and energy sectors. In this framework, analysing high-dimensional data presents some

✉ Fabrizio Maturo
fabrizio.maturo@unicampania.it

Rosanna Verde
rosanna.verde@unicampania.it

¹ Department of Mathematics and Physics, University of Campania Luigi Vanvitelli, Caserta, Italy

challenges, mainly due to the curse of dimensionality and the difficulty of dealing with data observed over time using conventional data analysis methods. Definitely, clustering and supervised classification techniques have opened new methodological perspectives in the learning from these huge amounts of complex data. In this context, Functional Data Analysis (FDA) has undergone significant developments over the past two decades. Indeed, FDA can deal with high dimensionality and also grasp additional information in the patterns of curves, for example, through derivatives or curvature (Ferraty and Vieu 2003; Ramsay and Silverman 2005).

The basic idea of FDA is to view the sets of scalar observations as a single object and then operate directly on the curves through smoothing and dimensional reduction techniques. Consequently, each statistical unit can be characterised by one or more functions depending on whether the one-dimensional or multidimensional case is considered (Ramsay and Silverman 2005). In both cases, the functions often have specific traits; discovering these characteristics is essential to acquiring additional information in statistical analysis. For example, many scholars have emphasised how investigating the behaviour of derivatives or curvature can be more attractive, in some contexts, than analysing the original curves (see.g. Cuevas 2014; Maturro et al. 2019, 2020).

The reference context of the present paper is the supervised classification of high-dimensional data through FDA. In particular, the goal is to create a classifier for a categorical response variable based on functional predictors, i.e. the scalar-on-function classification problem. Therefore, the primary idea is to extract information from the curves to get new features for the classification task. The further crucial aspect is based on how to merge the latter information with a functional classifier (see.g. Ramsay and Silverman 2005; Ferraty and Vieu 2006; Cuevas et al. 2007; Preda et al. 2007; Febrero-Bande and de la Fuente 2012). To improve the accuracy of the functional classifier by discovering additional information from data, B-splines and functional principal components are adopted. Indeed, they provide supplementary knowledge on the original curves and can be used as features for training a functional classification rule (see e.g. Febrero-Bande and de la Fuente 2012; Maturro and Verde 2022a). However, since the outcome is a categorical variable, this research suggests a strategy to discover extra information about the groups before training the functional classifier. Definitely, in real life, phenomena are often characterised by subpatterns. In other words, curves that belong to the same class and accordingly with the same label can be characterised by very distinct behaviours. For example, electrocardiogram signals of people affected by myocardial infarction may show different shapes over time. Creating a functional classifier overlooking this knowledge about subpatterns is undoubtedly a waste of information that can be used profitably to train the functional classifier.

Starting from the latter consideration, this study strives to improve classical functional classifiers' performance via an original two-phase clustering-classification strategy (hereafter denoted with "*Clustering and Train (C & T)*") that exploits information on subpatterns of the original groups. A first clustering step is performed to discover distinct clusters of curves among the same labelled classes. In the second step, supervised classification is performed by exploiting the extra information on the new subgroups derived from the first step. Thus, the basic idea is to get

additional knowledge in the training data to improve the performance of the final classifier.

The clustering in the first step is performed separately for each starting group. Therefore, even if a functional classifier is created using a number of classes higher than the original ones, each new subgroup is related to the original labelled classes. Hence, it is straightforward to bring the labels of the subgroups to those of the original groups to assess the performance of the trained functional classifier. This study concentrates on the Functional K-Means (FKM) algorithm because it provides optimal results in terms of homogeneity of the subgroups. However, different clustering methods and metrics or semi-metrics to compute the distance between curves can be adopted in the first step. Regarding the clustering process, several strategies can be used to determine the optimal number of subgroups (see e.g. Ramsay and Silverman 2005; Ramsay et al. 2009). To illustrate the proposal, this research refers to some functional classifiers, such as the functional k-nn (see e.g. García et al. 2015; Febrero-Bande and de la Fuente 2012; Jacques and Preda 2013), and the more recent functional random forest based on B-splines or functional principal components (see e.g. Yu and Lambert 1999; Maturo and Verde 2022a, b). Definitely, the proposed strategy can also be extended to other classifiers.

The remainder of this paper is as follows. Section 2 illustrates the so-called C & T procedure. Section 3 presents an application to a real data set concerning ECG data. Section 3.2 displays a simulation study with six different scenarios and compares classification methods with and without augmenting the number of classes. Finally, Sect. 4 ends the paper with a discussion and conclusions.

2 Material and methods

2.1 Functional data representation

Starting from time-series data, the FDA's basic idea is to work directly on the curves rather than the scalars given by the time observations. In other words, the focus shifts to the functions and their characteristics rather than to single temporal observations. This approach has several advantages. The first is undoubtedly an intrinsic dimensionality reduction due to the representation of data through fixed or data-driven basis systems (Ramsay and Silverman 2005). The second is to exploit additional information that the starting data do not highlight, e.g. derivatives, curvature, integrals, etc (Ferraty and Vieu 2006; Cuevas 2014). Furthermore, no particular assumptions are required, and it is possible to analyze data that also occur at irregular intervals. Finally, there is the theoretical possibility of observing the phenomenon in a much finer grid and, in the limit, to observe it at any fixed instant. Usually, the reference domain of the functions is time, but FDA can also be used in contexts where the domain is different (see e.g. Maturo et al. 2019).

In simple terms, FDA usually refers to those statistical issues where the available data consist of a sample of n functions, $x_1(t), x_2(t), \dots, x_N(t)$, defined on a compact interval. FDA takes some connections with those statistical issues, often referred to as inference in stochastic processes where the sample information is given by a

partial trajectory $x(t)$, $t \in [0, T]$ of a stochastic process $\{X(t), t \geq 0\}$ (Cuevas 2014). When a process $\{X(t), t \geq 0\}$ is monitored, one usually records the values in a discrete grid t_1, t_2, \dots, t_N . Hence, at the end, there is always a possibly high-dimensional vector observation $x(t_1), x(t_2), \dots, x(t_N)$ (Cuevas 2014).

Focusing our attention to the case of a Hilbert space with a metric $d(\cdot, \cdot)$ associated with a norm so that $d(x_1(t), x_2(t)) = \|x_1(t) - x_2(t)\|$, and where the norm $\|\cdot\|$ is associated with an inner product $\langle \cdot, \cdot \rangle$ so that $\|x(t)\| = \langle x(t), x(t) \rangle^{1/2}$, we can obtain as a specific case the space \mathcal{L}_2 of real square-integrable functions defined on τ by $\langle x_1(t), x_2(t) \rangle = \int_{\tau} x_1(t)x_2(t)dt$. Therefore, if $x(t) \in \mathcal{L}_2$, a basis function system is a set of known functions $\phi_j(t)$ that are linearly independent of each other and which span \mathcal{L}_2 (Ramsay and Silverman 2005).

The most standard technique to describe functions is to exploit a finite representation in a fixed basis system (Ramsay and Silverman 2005) as follows:

$$x_i(t) \approx \sum_{\omega=1}^{\Omega} c_{i\omega} \phi_{\omega}(t), \tag{1}$$

where $c_i = (c_{i1}, \dots, c_{i\Omega})^T$ ($i = 1, 2, \dots, N$) is the vector of coefficients describing the linear combination and $\phi_{\omega}(t)$ is the ω -th basis function, from a subset of $\Omega < \infty$ functions that can be used to approximate the whole basis expansion.

One of the most adopted techniques to express curves via a data-driven basis system is the Functional Principal Components (FPCs) decomposition, which leads to a dimensionality reduction whilst keeping the maximum portion of information from the starting data (Ramsay and Silverman 2005; Aguilera and Aguilera-Morillo 2013; Febrero-Bande and de la Fuente 2012). In this case, the functional data can be expressed as follows:

$$x_i(t) = \sum_{k=1}^K v_{ik} \xi_k(t), \tag{2}$$

where K is the total number of FPCs, v_{ik} is the score of the generic FPC ξ_k of the i -th function $x_i(t)$ ($i = 1, 2, \dots, N$). By trimming this representation in terms of the first p FPCs, it is possible to get an approximation of the sample curves, whose explained variance is given by $\sum_{k=1}^p \lambda_k$, where λ_k is the variance of the k -th functional principal component. The most significant benefit of the latter procedure is that it captures the primary characteristics of the data using just a smaller set of uncorrelated FPCs¹

¹ Centering the curves so that the sample mean is equal to 0, we define the covariance function $v(t, r) = N^{-1} \sum_{i=1}^N x_i(t)x_i(r)$ and, to find the principal components weight function ξ_k , we maximize the variance by solving the characteristic equation

$$\int_{\tau} v(t, r) \xi_k(t) dt = \lambda_k \xi_k(t),$$

s.t.

$$\int_{\tau} \xi_k(t)^2 dt = 1 \quad \text{and} \quad \int_{\tau} \xi_k(t) \xi_l(t) dt = 0, \quad \text{for } l \neq k.$$

2.2 Unsupervised and supervised classification in the FDA context

As in the traditional (non-functional) statistical context, one can distinguish between the unsupervised and supervised classification. Unsupervised classification coincides with clustering and is therefore always based on creating groups of curves that are similar to each other within groups and dissimilar as much as possible between groups. The term “unsupervised” naturally highlights that there is no prior knowledge of class labels or even the number of possible classes. Instead, in functional supervised classification, the labels of each curve are known a priori, and they are used to train a functional classifier, i.e. a classification rule that can be exploited to predict the unknown value of the grouping variable of any new curves. The literature on functional classification, in recent decades, has extended much of the classical statistics to the case in which objects of study are functions.

Both in the context of clustering and supervised classification, proximity measures among statistical units play a critical role because, according to different chosen distances, contrasting results can be achieved. The choice of a proximity measure depends on the nature of the data and the purpose of the specific research. In the context of FDA, different metrics and semi-metrics can be used; however, limiting our consideration to the case of the \mathcal{L}_2 -space, the most ordinarily employed proximity measures between functional elements are the \mathcal{L}_2 -distance, and the semi-metrics of the FPCs or derivatives (Ramsay and Silverman 2005; Ferraty and Vieu 2006; Febrero-Bande and de la Fuente 2012).

The \mathcal{L}_2 -distance is given by:

$$\|x_1(t) - x_2(t)\|_2 = \sqrt{\int_{\tau} [x_1(t) - x_2(t)]^2 dt}, \tag{3}$$

where the observed points on each curve are equally spaced. Instead, the semi-metric of the FPCs is given by:

$$d_2(x_1(t), x_2(t)) \approx \sqrt{\sum_{k=1}^K (v_{1,k} - v_{2,k})^2 \|\xi_k\|}, \tag{4}$$

where $v_{i,k}$ is the coefficients of the expansion, and ξ_k is the k -th orthonormal eigenvector. Often, the semi-metric of the r -order derivatives of two curves could also be considered because it furnishes compelling knowledge depending on the scope of the study (Ramsay and Silverman 2005; Febrero-Bande and de la Fuente 2012).

Footnote 1 (continued)

where the i -th FPCs scores are given by $v_{ik} = \int_{\tau} x_i(t) \xi_k(t) dt$, $i = 1, \dots, N$, and λ_k is the variance explained by the k -th functional principal component.

2.2.1 The functional K-means (FKM)

There are many strategies available in the functional clustering literature. For example, Jacques and Preda (2013) presents an interesting review and also Febrero-Bande and de la Fuente (2012) implement many methods in the R package *fda.usc*. In the following, this research focuses on the FKM as a possible unsupervised classification technique for functional data, but keeping in mind that the approach can be extended to other clustering strategies.

The fundamental idea of FKM is to look for a partition for which the variability within clusters is minimized. Starting from N functional observations, this method pursues to group functional data into $G \ll N$ groups, C_1, C_2, \dots, C_G to minimize the within-cluster sum of squares. The initial phase of this iterative process involves fixing G initial functional centroids, $\psi_1^{(0)}(t), \dots, \psi_G^{(0)}(t)$. Subsequently, each curve is assigned to the cluster whose centroid, at the prior iteration ($\delta - 1$) is the closest according to the desired metric:

$$C_g^{(\delta)} = \arg \min_{g \in 1, \dots, G} d_2 \left(x_i(t), \psi_g^{\delta-1}(t) \right), \quad \delta = 1, \dots, \Delta, \quad (5)$$

where Δ is the maximum number of stages of the algorithm. Once all the curves have been allocated in a group, the cluster functional means are updated as follows:

$$\psi_g^\delta(t) = \sum_{x_i(t) \in C_g} \frac{x_i(t)}{n_g}, \quad (6)$$

where n_g is the number of functions in the g -th cluster, C_g (Febrero-Bande and de la Fuente 2012; Fortuna et al. 2018; Maturó et al. 2020). The procedure ends when the curves no longer change group or the maximum number of predetermined iterations is reached.

The choice of the number of groups, as in any clustering method, is significant for identifying patterns within the original data. By expressing the curves of the training set through a basis system, it is possible to exploit the classic methods for determining the number of groups. In fact, working on the scores of the FPCs or B-spline, the so-called “direct methods” and also the statistical testing approaches can help to choose the number of groups. The former strategies optimise a criterion, such as the within-cluster sums of squares or the average silhouette. The corresponding procedures are the elbow and silhouette methods, respectively. Instead, the latter compares evidence against the null hypothesis, e.g. the gap statistic. In the literature, there are many other techniques available for choosing the number of groups, but a complete review is out of the scope of this paper because the choice of the number of groups is secondary to the main strategy. Consequently, in the following, this study concentrates only on the silhouette method extended to the FPCs and B-splines scores to identify the suitable number of subgroups to be used for the FKM initialisation.

The average silhouette determines how well each curve lies within its cluster and can be computed for different values of G . The optimal number of clusters G^* is the

one that maximizes the average silhouette over a range of possible values for G . The silhouette for the i -th curve is computed as follows:

$$S(i) = \frac{b(i) - a(i)}{\max(b(i), a(i))}, \tag{7}$$

where: $a(i) = \frac{1}{n_g - 1} \sum_{j \in C_g} d(x_i(t), x_j(t))$ is the mean distance of the i -th curve with respect to all the functions belonging to the same cluster C_g (with n_g the number of functions in the class C_g ; -1 because is not included x_i) and $b(i) = \min_{l \neq g} d(x_i(t), x_l(t))$ (with $x_l \notin C_g$) is the minimum distance between the i -th curve and all the curves belonging to the other classes.

2.2.2 The functional K-NN (FKNN) and functional random forest (FRF)

In the FDA context, there are many strategies of supervised classification (see e.g. Preda et al. 2007; Febrero-Bande and de la Fuente 2012; Chang et al. 2014; Baíllo et al. 2018; Mousavi and Sørensen 2018; Baíllo and Cuevas 2008; Maturó and Verde 2022a). To illustrate the proposal, in the second step, this research concentrates on the Functional K-NN (FKNN) (Febrero-Bande and de la Fuente 2012) and the Functional Random Forest (FRF) (Breiman 2004). Nevertheless, the suggested approach could be extended to further functional classifiers.

In the functional classification framework, the aim is to forecast the class (or label) of an observation x taking values in a separable metric space (χ, d) . Hence, our strategy is designed for functional data of the form $\{y_i, x_i(t)\}$, with a predictor curve $x_i(t)$, $t \in \tau$, and y_i being the categorical response, observed for $i = 1, \dots, N$. The classification of a new observation x from X is carried out by constructing a mapping $f : \chi \rightarrow \{1, \dots, H\}$, with H being the total number of categories of the response variable Y . The so-called “*classifier*” maps x into its predicted label with a probability of error given by $P\{f(X) \neq Y\}$.

Given a sample X , the aim is to estimate the posterior probability of belonging to each group C_h :

$$p_h(X) = P(y = C_h \mid x = X), \tag{8}$$

where $h = 1, \dots, H$ denotes the different modalities of Y .

The classification rule consists in assigning a new curve to that group with the maximum posterior probability.

$$\hat{y} = \arg \max \hat{p}_h(X). \tag{9}$$

The estimate of the posterior probability $p_h(X)$ can be calculated using different classifiers such as the FKNN or FRF that are implemented in the following (see e.g. Ramsay and Silverman 2005; Febrero-Bande and de la Fuente 2012; Maturó and Verde 2022b).

The FKNN classifier is a non-parametric supervised classification approach for functional data; it is probably the simplest and most used algorithm for classifying curves based on the classes of the “ k ” curves in the training set which are

closest to the one considered. Despite being a very simple classifier, by choosing a suitable value for the parameter k , it is a very performing algorithm in terms of accuracy (Febrero-Bande and de la Fuente 2012).

On the other hand, the FRF is an extension of the classical random forest to the FDA context (Breiman 2004). Starting from a single Functional Classification Tree (FCT) trained using the scores obtained via a selected functional representation technique, it is possible to build an ensemble of weak classifiers whose final result is very powerful in terms of accuracy and variance reduction.

Each FCT of the forest consists of recursive binary separations of the feature space into rectangular regions (nodes) composed of sets of curves $x_i(t) \in X$. In the building procedure, an optimal binary partition is implemented at each phase of the algorithm, based on optimising the cost criterion. Typically, the latter regards the reduction of the impurity of the node via the Gini index $G = 1 - \sum_h f_{rh}^2$, where f_{rh} denotes the proportion of training curves in the r -th region that are from the h -th class, and H is the number of categories of Y , or the Shannon-Weiner index $E = -\sum_h f_{rh} \ln f_{rh}$, where f_{rh} is the proportion of training observations in the r -th region that are from the h -th class (Hastie et al. 2009; Therneau and Atkinson 2019). The algorithm starts with the entire functional data set and continues until terminal nodes (leaves) are obtained (Maturò and Verde 2022a).

The reason of shifting from FCTs to FRF is similar to the non-functional context, that is, to lower the variability of estimates due to the presence of correlated FCTs (Breiman 2004; Hastie et al. 2009; James et al. 2013). Effectively, FRF creates many FCTs on B bootstrap replicates of the original dataset, decorrelating the FCT. For this purpose, a random sample of m features is considered at each split in such a way that FCTs are less dominated by the same predictors (a detailed description of this procedure is available in (Maturò and Verde 2022a)). Hence, at each division in the FCTs, the algorithm does not consider most of the available FPCs (or B-splines). Thus, on average, $\frac{K-m}{K}$ will not even be considered in the splitting procedures. A general rule of thumb can be to choose, as the size of the subset of FPCs (or B-splines), a value of $m \approx \sqrt{K}$. Because each FCT has its forecast label of the class, a new curve is assigned according to the so-called “majority vote” criterion (Breiman 2004).

Concerning the assessment of the functional classifiers’ accuracy, different strategies are available as in the non-functional framework. Indeed, both for FKNN and FRF, it is possible to exploit cross-validation, bootstrap, or validation test set (Hastie et al. 2009; James et al. 2013). In the FDA context, the use of a functional test set is of particular appeal because the test functions must be described according to the same basis system used to define the functional training set. If a fixed basis system is used to express the curves in the training set, the test functions can be represented by employing the same fixed basis system, e.g. with the same number and order of B-splines. On the contrary, if a data-driven basis system is used, the test curves x_s , ($s = 1, \dots, S$), must be projected onto the FPCs space generated by the training curves in order to get the appropriate scores as follows:

$$v_{sk} = \langle x_s^c, \xi_k \rangle = \int_{\tau} x_s^c(t) \xi_k(t) dt \quad s = 1, \dots, S, \quad (10)$$

where the weight functions ξ_k^t 's are obtained executing the FPCs on the training set X , x_s^c are the centered curves of the test set (obtained subtracting the sample mean function of the training samples), and S is the total number of functions in the test set.

2.3 The “clustering and train” (C &T) method

The main goal of the proposed strategy is to enhance the accuracy of the functional classifier by generating extra information on possible subpatterns of the original classes before training a functional classifier. With this aim, an original two-phase classification technique, namely “Clustering and Train” (C &T) is introduced. The latter combines unsupervised (first phase) and supervised (second phase) classification techniques in the FDA framework. An outline of the C &T procedure is illustrated in Algorithms 1 and 2.

The preliminary step of the procedure is the representation of high-dimensional time series into functional objects through the classical FDA techniques. Accordingly, the first decision involves choosing the basis system. If a fixed basis system is used, the training and test set can be easily represented employing the same number of basis with the same order. If, on the other hand, a data-driven basis system is adopted, the test set curves must be projected into the space generated by the FPCs generated by the training set functions. Once the curves are represented through a linear combination of basis functions, it is possible to extract the scores (or the coefficients) to be used as features.

The unsupervised classification phase involves critical choices such as the clustering method, number of subgroups, and metric or semi-metric to evaluate the similarity between functional data. It is worth nothing that the clustering procedure is applied separately for each original group of curves in the training set.

Once the subclasses have been identified, it is possible to move on to the second phase, which consists of supervised classification with an augmented number of classes. Therefore, it is necessary to choose a functional classifier and the possible values of its hyperparameters, since there are numerous classifiers in the FDA literature. The performance of the C &T strategy, in terms of accuracy, can be evaluated using the training set via cross-validation and bootstrap, or adopting a functional test set. However, the accuracy of the functional classifier must be assessed after tracing the predicted classes to the original classes. Having carried out the clustering separately for each original group, bringing the subclasses back to the original classes is immediate. In the following, this study refers to the test set approach for assessing the accuracy of the FKNN and FRF classifiers.

The aforementioned procedure usually improves the performance of the functional classifier, in particular in cases in which the original groups are composed by subpatterns. Definitely, the C &T algorithm can be generalised to different functional classifiers by replacing the “5: STEP 2”.

Algorithm 1 - Clustering and Train algorithm using the FRF Classifier

- 1: **INPUT DATA:** $X = \{\mathbf{x}_1, \dots, \mathbf{x}_i, \dots, \mathbf{x}_N\}; Y = \{C_1, \dots, C_h, \dots, C_H\}$
 - 2: **Basis transformations** $f(x_i)$

$$\begin{array}{l} \xrightarrow{\text{FPCs}} \xi_k(t) \\ \xrightarrow{\text{B-splines}} \phi_\omega(t) \end{array}$$
 - 3: **STEP 1 - Clustering algorithm:** partition of $X(t) | Y = C_h$ into sub-sets: $x_i(t) \in C_h$ ($h = 1, \dots, H$) in a fixed number of subclasses
 - Repeat:**
 - $h \leftarrow 1$
 - Fix the number of clusters G_h and the maximum number of iterations
 - Clustering C_h in G_h subclasses
 - Select the optimal number of clusters G_h^* .
 - Compute the Silhouette index $S_h(i)$ [or Elbow or Gap Statistics].
 - $h \leftarrow h + 1$
 - until:** $h = H$
 - 4: **return:** subpatterns of the original classes
Augmenting class number of $Y \rightarrow Y^* = \sum_{h=1}^H G_h^*$
 - 5: **STEP 2 - FRF Classifier:**
 - FPCs'** scores features: $v_1(t), \dots, v_k(t), \dots, v_K(t)$;
 - [OR **B-splines'** coefficients features: $c_1(t), \dots, c_\omega(t), \dots, c_\Omega(t)$];
 - response categorical variable: Y^*
 - FRF procedure:**
 - Fix L = maximum number of FCTs
 - Repeat:**
 - $nt \leftarrow 1$
 - Bootstrap of B samples X_B (of size N)
 - Selection of m FPCs $\xi_k(t)$ for each split
 - Built B **FCTs**
 - Split criterion: **G** (Gini's index) [or **E** (Shannon's Entropy)]
 - until:** $nt \leftarrow L$
 - 6: **return: Classifier** F
 - 7: Assign $x_i(t)$ to Y^* via majority vote and bring the predicted subclasses to the original ones in Y
 - 8: **STEP 3 - Assessment of the Classifier**
 - Accuracy evaluation on the training set using bootstrap [or cross-validation]
 - return:** Confusion matrix and accuracy evaluation using the training set
 - 9: **STEP 4 - Validation**
 - Select a **Test set** as a new sample from the same population of curves of the training set: $x_s \in X$
 - return:** Confusion matrix and accuracy evaluation using the test set
 - Transform $x_s(t)$ according to the **FPCs** [OR **B-splines**] carried out on the training set
 - 10: **Prediction of the test set**
 - return:** Accuracy evaluation
 - 11: **STOP**
-

Algorithm 2 - Clustering and Train algorithm using the FKNN Classifier

- 1: **INPUT DATA:** $X = \{\mathbf{x}_1, \dots, \mathbf{x}_i, \dots, \mathbf{x}_N\}; Y = \{C_1, \dots, C_h, \dots, C_H\}$
 - 2: **Basis transformations** $f(x_i)$: FPCs [or B-splines]
 - 3: **STEP 1 - Clustering algorithm:** partition of the $X(t) | Y = C_h$ sub-sets: $x_i(t) \in C_h (h = 1, \dots, H)$ in a fixed number of subclasses
- like in Alg.1
 - 4: **return:** subpatterns of the original classes
Augmenting class number in $Y \rightarrow Y^*$
 - 5: **STEP 2 - FKNN Classifier:**
Fix $K =$ number of the neighbour curves
for $i=1$ **to** N
Calculate the semi-metric of FPCs [or \mathcal{L}_2 distance or semi-metric based on B-spline basis]
Take the K nearest neighbours
Count the number of the curves in each class
Assign $x_i(t)$ to the class with the highest $p_h(X)$
end
 - 6: **return:** class prediction of Y^*
 - 7: Bring the subclasses Y^* to the original ones in Y
 - 8: **STEP 3 - Assessment of the Classifier**
Accuracy evaluation on training set by bootstrap
return: Accuracy on training set
 - 9: **STEP 4 - Validation**
- like in Alg. 1
 - 10: **Prediction of the test set**
return: Accuracy evaluation
 - 11: **STOP**
-

3 Applications and results

3.1 Application to the ECG200 dataset

This section aims to illustrate the methodology proposed on a dataset concerning the electrocardiogram signals (ECG). The ECG200 dataset was presented by R. Olszewski at Carnegie Mellon University in 2001 as part of his work “*Generalized feature extraction for structural pattern recognition in time-series data*” (Olszewski 2001). The dataset is worldwide adopted to test new classifiers, and the existing world record, in terms of classification accuracy, is 89.05%. Each sequence traces the electrical activity recorded during one heartbeat. The two classes are Normal Heartbeat (NH) and Myocardial Infarction (MI). Both the training and test sets are composed of 100 signals. The data is freely obtainable at www.timeseriesclassification.com (Bagnall et al. 2021). Our purpose is to forecast whether a new patient is healthy or diseased.

Figure 1 illustrates the original ECGs in the training and test set. The original signals are time records, and the observations are joined through a simple graphical interpolation. The red signals identify healthy patients, that is, those with a regular heartbeat. Black signals display sick patients, i.e. those suffering from

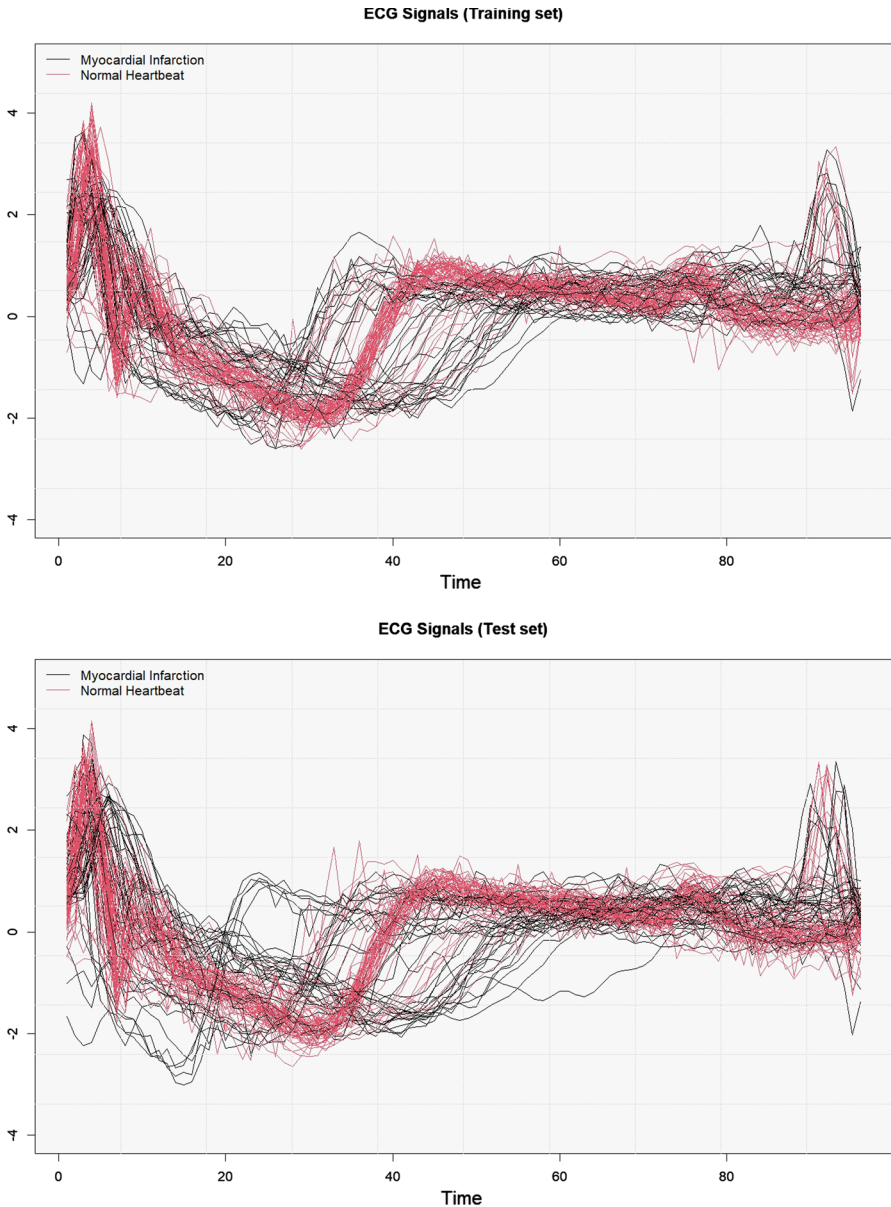


Fig. 1 ECG signals of the training and test set (ECG200 dataset). The R packages *fda* (Ramsay et al. 2022) and *fda.usc* (Febrero-Bande and de la Fuente 2012) are used to represent functional data

myocardial infarction. The charts clearly show that the signals of sick patients are characterized by subpatterns. In other words, not all patients suffering from myocardial infarction have similar electrocardiogram trends.

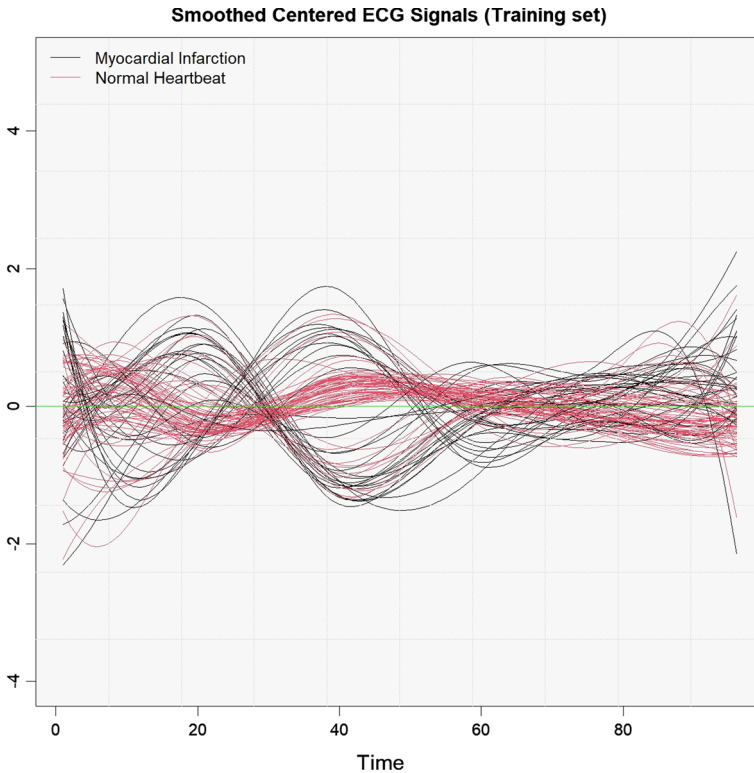


Fig. 2 Centered smoothed curves of the training set (ECG200 dataset)

Figure 2 displays the centered smoothed curves of the training set. This picture highlights the functional differences between healthy and diseased people. Also, it helps to understand Fig. 3 which presents the FPCs decomposition of the curves in the training set. The first four FPCs explain about 80% percent of the variability. However, we cannot be satisfied with explaining 80% because, in the perspective of supervised classification, the FPCs that explain little variability are essential in ensuring a good accuracy of the classifier.

Figure 4 exhibits the average silhouette to measure the quality of clustering for the different number of groups. The two graphs in Fig. 4 highlight the desired number of subgroups for the Myocardial Infarction (MI) and Normal Heartbeat (NH) original groups, respectively. The MI group can be decomposed into two subgroups, whereas the NH group furnishes three different patterns.

Figure 5 illustrates the FKM results in terms of final functional centroids. The latter remark that the original groups are characterized by clear subpatterns.

Figures 6 and 7 show the FCTs trained using the FPCs and B-splines, respectively. The latter do not use the C & T technique, and thus the leaves are given by the original labels of the outcome, i.e. MY and NH. The R packages *rpart* (Therneau and Atkinson 2019) and *rpart.plot* (Milborrow 2021) are used to

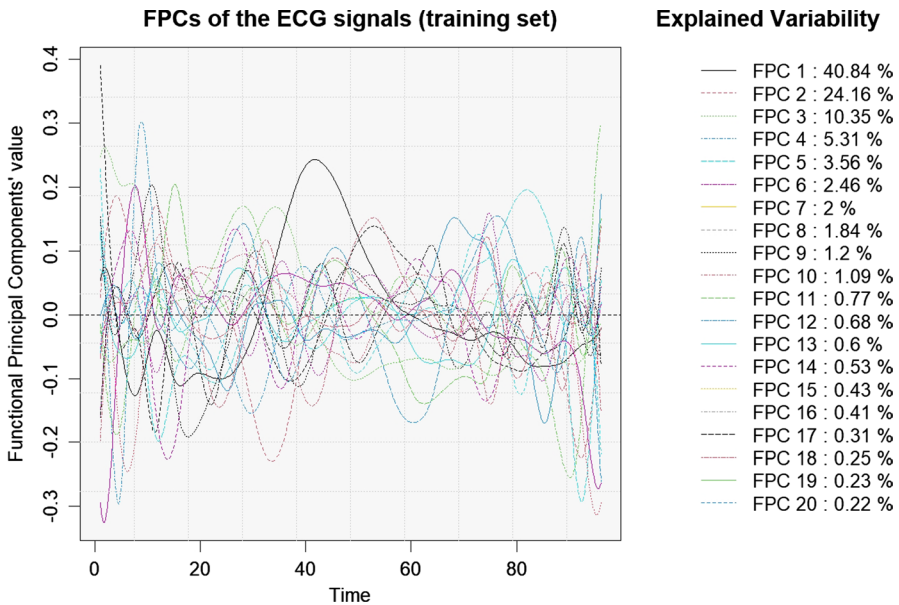


Fig. 3 FPCs decomposition of the training set curves (ECG200 dataset).

represent FTCs with the appropriate adjustments due to the original methodology. The splitting criterion adopted in this study is based on the Gini index.

Figures 8 and 9 offer the FCTs trained using the FPCs and B-splines approaches with the C & T method, respectively. Therefore, the terminal nodes are no longer represented by the original labels but by the subgroups of the original labels; in fact, the number of classes to predict has increased to five. The single FCT has only an explanatory role in the methodology because, effectively, FRF deals with an ensemble of FCTs.

Figures 10 and 11 offer the results of the FRF-B-splines and FRF-FPCs, respectively. Both the pictures compare the FRF without the C & T approach and FRF performed via the C & T technique. The accuracy is computed based on the test set and is plotted as the forest size varies. The results of the FRF-B-splines always consider a fixed number of basis functions. Instead, the FRF-FPC deals with a number of FPCs from 2 to 20. Consequently, two pieces of information can be exploited in the performance assessment of the FRF-FPCs classifier. The first is represented by the maximum value reached by the accuracy (dotted curves). On the other hand, the average accuracy value gives the second fascinating information (solid curves which provide the average accuracy using a different number of FPCs given the size of the forest). The latter information is definitely the most important because it is little affected by fluctuations due to chance. Figures 10 and 11 highlight that the FRF-B-splines and FRF-FPCs performed via the C & T technique provides extraordinary results on this dataset. In fact, the previous record (89.05% accuracy) is repeatedly beaten with both the classifiers. Indeed, the FRF-FPCs classifier achieves 94% accuracy for many forest sizes. Another aspect worthy of being highlighted is

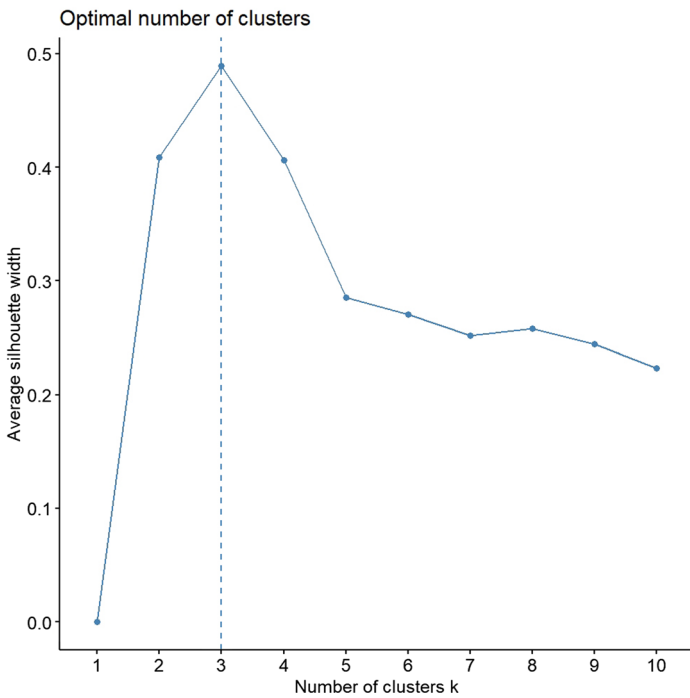
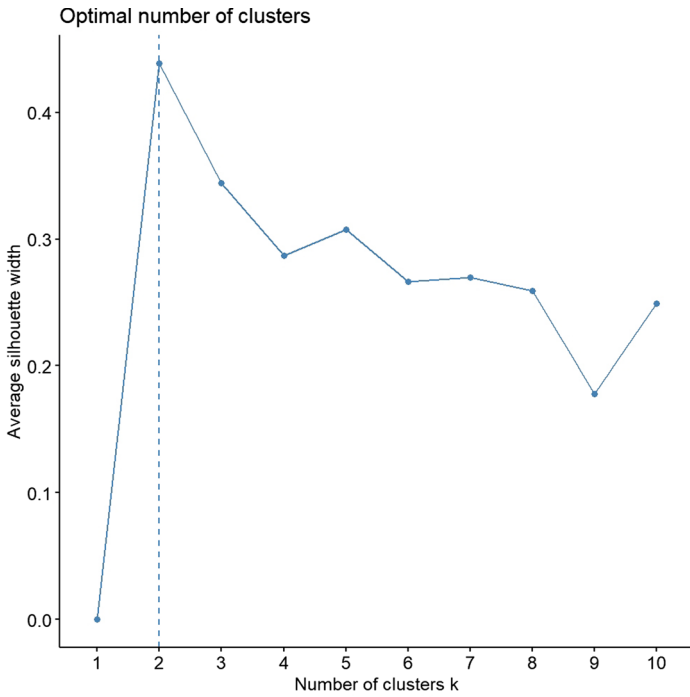


Fig. 4 Number of subgroups selection for the Myocardial Infarction and Normal Heartbeat original groups (ECG200 dataset)

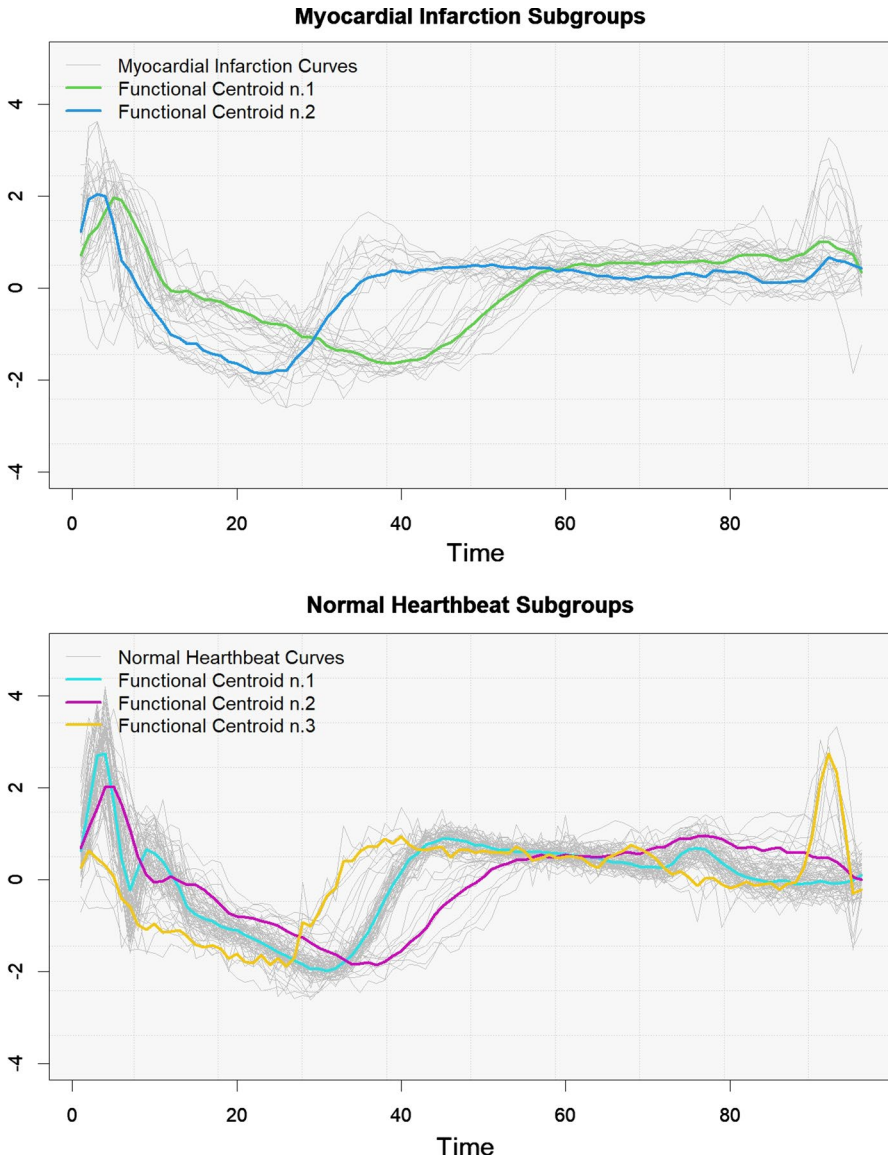


Fig. 5 Functional centroids of the Myocardial Infarction and Normal Heartbeat original groups according to the FKM clustering procedure (ECG200 dataset)

that, in Fig. 11, the mean accuracy of the FRF-FPCs via the C & T is systematically above that one of the classical functional classifier, and also higher than the previous 89.05% accuracy record.

Figure 12 describes the results of C & T adopted employing FKNN in the supervised classification phase. A comparison between the classical FKNN and FKNN with augmented labels is provided. Also in this circumstance, the C & T method

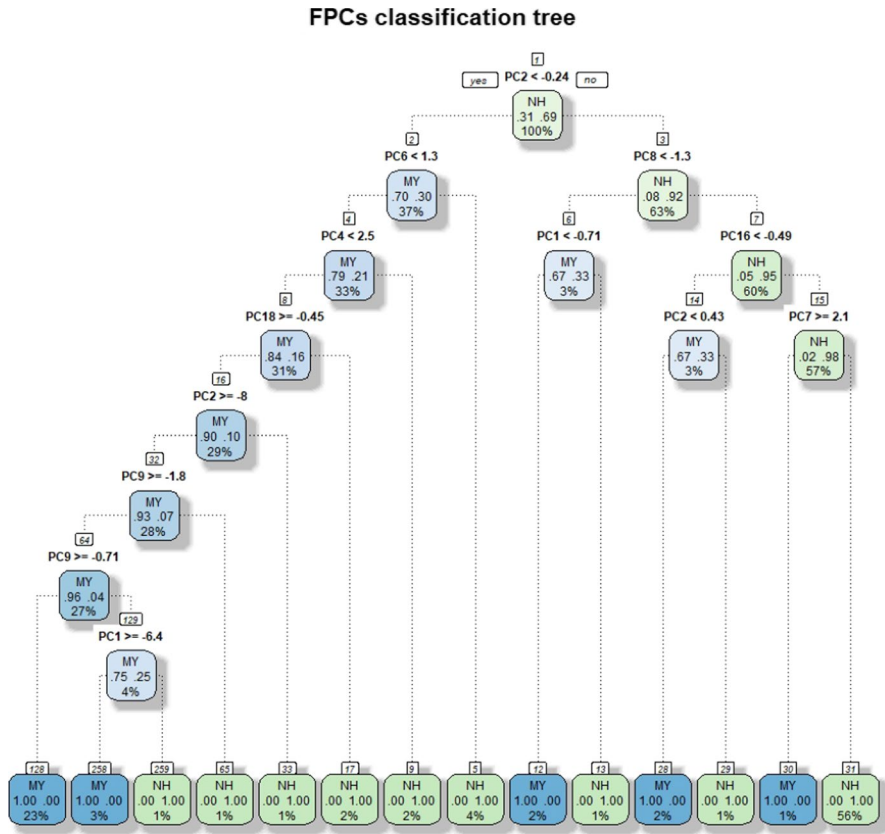


Fig. 6 FPCs classification tree (ECG200 dataset)

enhances the accuracy of the classical FKNN. Precisely, with 3-NN, 92% accuracy is obtained.

3.2 Simulation study

To exhibit the performance of the novel strategy, diverse $X_i(t)$ models suggested in previous studies are proposed (Cuevas et al. 2007; Preda et al. 2007; Taiwo Ojo et al. 2021). Particularly, there are six scenarios in which the first four take into account the case of a binary classification problem, and the last two consider three and four classes, respectively. In each scenario, 100 functions for any group are generated. Consequently, in the two-class classification problems, there are 200 curves, while in the last two scenarios, there are 300 and 400 curves, respectively. The different simulations are obtained using the following six scenarios.

Simulation 1. Group 1 is generated by the model $X_i(t) = \mu t + e_i(t)$ and group 2 is generated by the model $X_i(t) = \mu t + qk_i I_{T_i \leq t} + e_i(t)$ where $t \in [0, 1]$, $e_i(t)$ is a Gaussian process with zero mean and covariance of the form

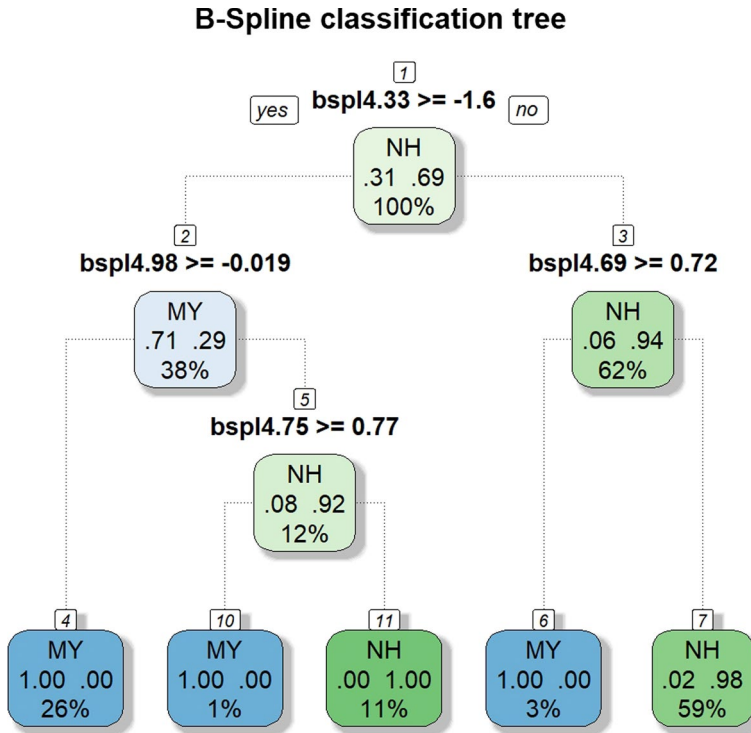


Fig. 7 B-spline classification tree (ECG200 dataset)

$\gamma(s, t) = \alpha \exp \{-\beta | t - s |^\nu\}$, $k_i \in \{-1, 1\}$ with $P(k_i = -1) = P(k_i = 1) = 0.5$, I is an indicator function, q is a constant controlling how far the curves in group 2 are from the mass of group 1, and T_i is a uniform random variable in an interval $[a, b] \subset [0, 1]$. This simulation allows us to get two groups that differ in their magnitude only in a specific part of the time domain. Figure 13(1) shows the simulated data obtained fixing $\mu = -2$, $q = -2$, $a = 0.2$, $b = 0.8$, $\alpha = 0.1$, $\beta = 1$, and $\nu = 0.9$.

Simulation 2. We take into account the following two functional data generating models to get two groups, which differ mainly due to their amplitude. The main model is $X_i(t) = a_{1i} \sin \pi + a_{2i} \cos \pi + e_i(t)$. To obtain group 2 we refer to the model $X_i(t) = (b_{1i} \sin \pi + b_{2i} \cos \pi)(1 - u_i) + (c_{1i} \sin \pi + c_{2i} \cos \pi)u_i + e_i(t)$ where $t \in [0, 1]$, $\pi \in [0, 2\pi]$, a_{1i}, a_{2i} follows a uniform distribution in an interval $[a_1, a_2]$, b_{1i}, b_{2i} follows uniform distribution in an interval $[b_1, b_2]$, c_{1i}, c_{2i} follows uniform distribution in an interval $[c_1, c_2]$, u_i follows Bernoulli distribution, and $e_i(t)$ is a Gaussian process with zero mean and covariance function of the form $\gamma(s, t) = \alpha \exp \{-\beta | t - s |^\nu\}$. Figure 13(2) shows the simulated data obtained fixing $a_{1i} = 2$, $a_{2i} = 29$, $b_{1i} = 1.5$, $b_{2i} = 23$, $c_{1i} = 1$, $c_{2i} = 15$, $\alpha = 12$, $\beta = 0.5$, and $\nu = 1$.

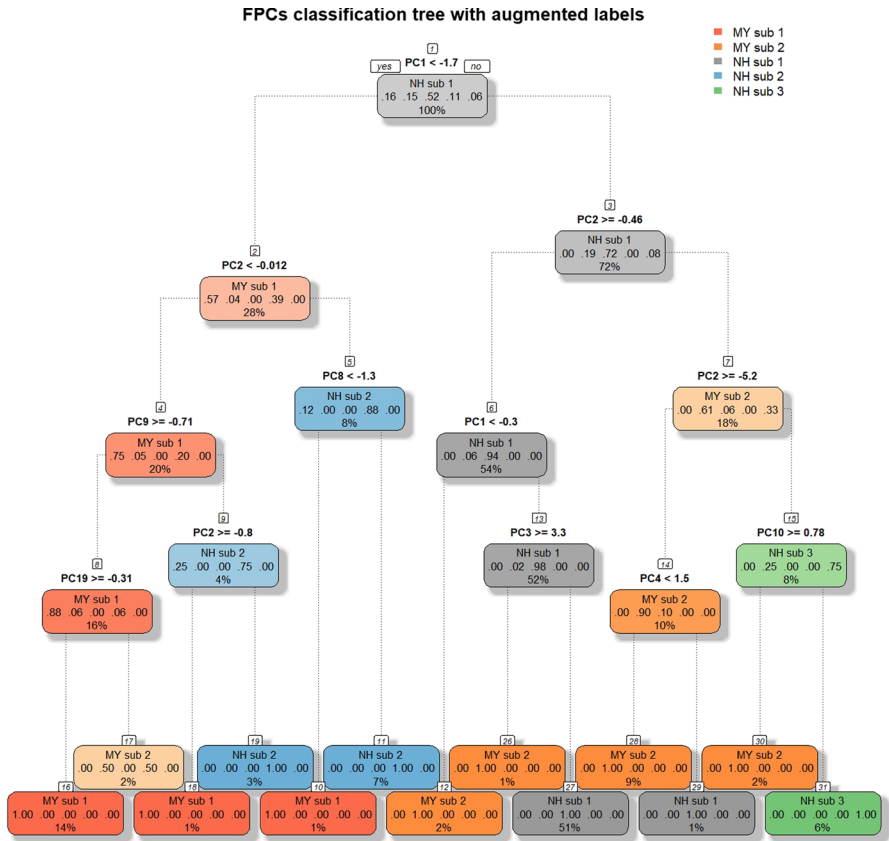


Fig. 8 FPCs classification tree with augmented labels (ECG200 dataset)

Simulation 3. We consider the following two functional data generating models to obtain two groups, which differ mainly according to their amplitude. The main model is of the form $X_i(t) = \mu t + e_i(t)$. To obtain group 2 we refer to the model $X_i(t) = \mu t + k \sin(r\pi(t + \theta)) + e_i(t)$ where $t \in [0, 1]$, and $e_i(t)$ is a Gaussian process with zero mean and covariance function of the form $\gamma(s, t) = \alpha \exp\{-\beta |t - s|^\nu\}$, θ is uniformly distributed in an interval $[a, b]$, and k, r are constants. Figure 13(3) shows the simulated data obtained fixing $\mu = -12, a = 0, b = 0.99, r = 1, k = 5, \alpha = 1, \beta = 0.3,$ and $\nu = 0.6$.

Simulation 4. We use two functional data generating models to get two groups, which have a slight dissimilarity in magnitude and shape in a portion of the time domain. Group 1 is achieved employing the model $X_i(t) = \mu t + e_i(t)$ whereas group 2 is generated by the model $X_i(t) = \mu t + (-1)^u q + (-1)^{(1-u)} \left(\frac{1}{\sqrt{r}}\right) \exp(-z(t - v)^w) + e_i(t)$, where $t \in [0, 1], e_i(t)$ is a Gaussian process with zero mean and covariance function of the form $\gamma(s, t) = \alpha \exp\{-\beta |t - s|^\nu\}, u$ follows a Bernoulli distribution

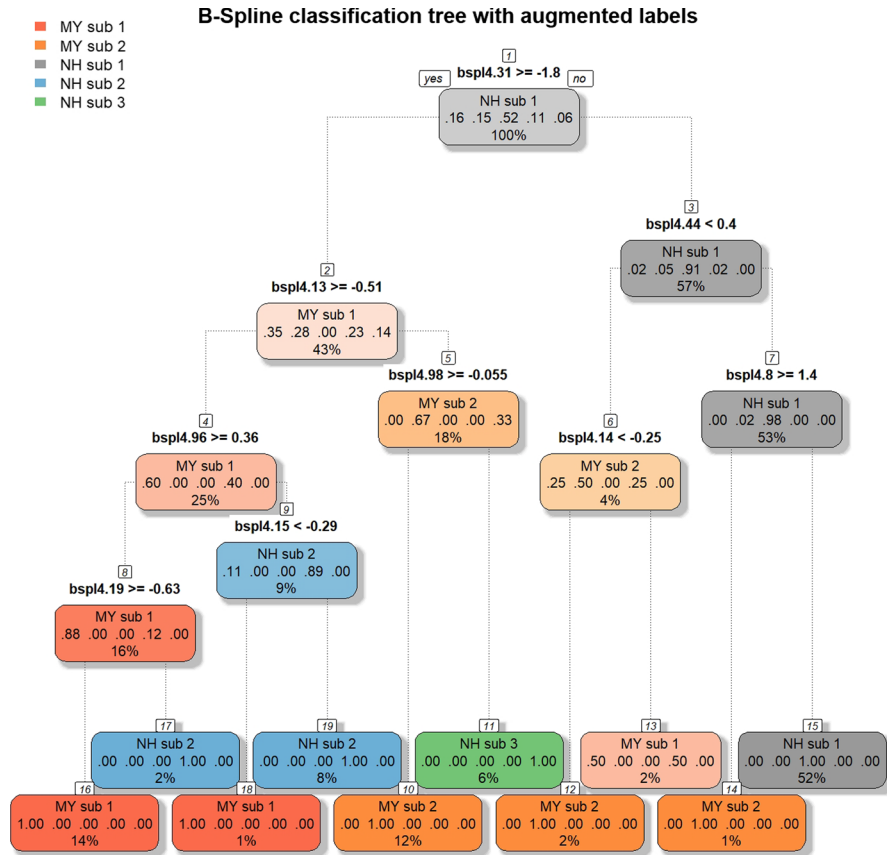


Fig. 9 B-spline classification tree with augmented labels (ECG200 dataset)

with $P(u = 1) = 0.5$, q, r, z and w are constants, v follows a uniform distribution in $[a, b]$. The two sets of curves are got specifying the following parameters: $\mu = 0, q = 2, a = 0, b = 0.55, \alpha = 0.51, \beta = 1, v = 1, w = 2, r = 0.02, z = 3$, and $w = 2$. Figure 13(4) displays the simulated curves.

Simulation 5. Simulation 5 exploits the model considered in Simulation 1 with suitable adjustments to the three-classes classification problem. Specifically, Fig. 13(5) shows the simulated data obtained fixing $\mu = -14, q = 3, a = 0.6, b = 0.75, \alpha = 2, \beta = 1$, and $v = 0.5$ for Groups 1 and 2, and $\mu = -14, q = 5, a = 0.2, b = 0.95, \alpha = 2, \beta = 1$, and $v = 0.5$ for Group 3.

Simulation 6. Simulation 6 adopts the model used in Simulation 4, adjusting to the four-classes classification undertaking. Specifically, Fig. 13(6) displays the simulated data given by $\mu = 0, q = 1.8, a = 0.45, b = 0.45, \alpha = 1, \beta = 1, v = 1, w = 2, r = 0.02$, and $z = 90$ for Groups 1 and 2, and $\mu = -2, q = 1.8, a = 0.15, b = 0.15, \alpha = 0.8, \beta = 0.8, v = 1, w = 4, r = 0.01$, and $z = 90$ for Groups 3 and 4.

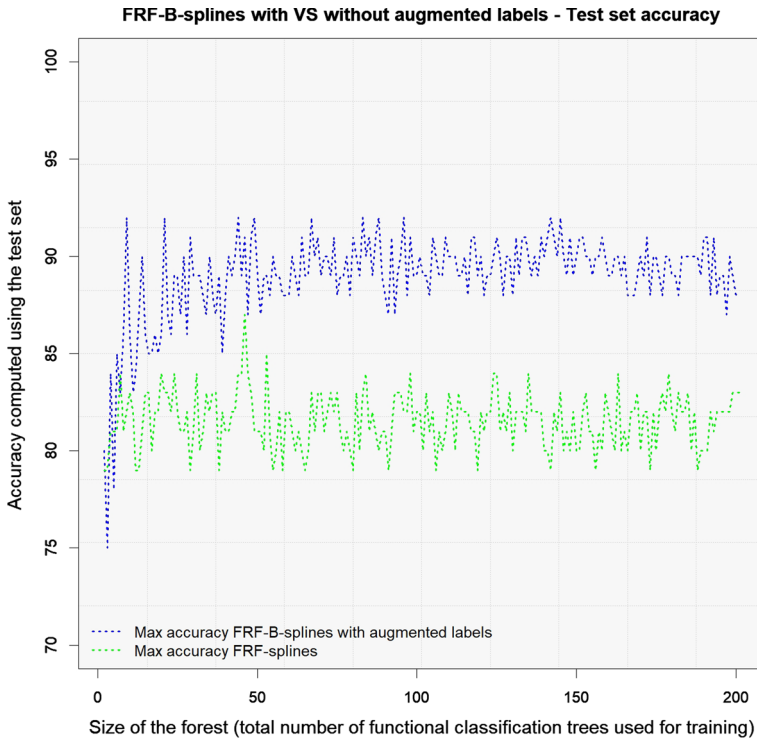


Fig. 10 Comparison of max accuracy computed on the test set between the classical FRF-B-spline and FRF-B-spline with augmented labels classifiers (ECG200 dataset). The mean accuracy is not considered in the FRF-B-spline classifier because a fixed number of B-splines is used, and thus no average is available for a given size of the forest

We compare the classical functional classifiers for each simulation that employed the FRF-FPCs, FRF-B-splines, and FKK operating the C & T approach. The accuracy is calculated using the test set, following the same scheme and reasoning adopted for the ECG200 dataset. Accordingly, for each scenario, three detailed comparisons are offered.

Figure 14b and c show that in Scenario 1, the C & T method enhances the classification accuracy for both the FRF-FPCs and FKNN approaches. Rather, Fig. 14a exhibits that, using B-splines, there is no improvement for the classical FRF.

In scenario 2, Fig. 15a, b, and c highlight the excellent results of the proposed approach. Specifically, the average accuracy of the FRF-FPCs classifiers is systematically exceeding the mean of the functional classifier not using the two-phase process (especially when the forest size exceeds 80 FCTs).

Scenario 3 is a straightforward classification task, and consequently, the results of all the methods are very high in terms of accuracy. Thus, the performances of the classical approaches are identical to those of the novel technique (Fig. 16a, b and c).

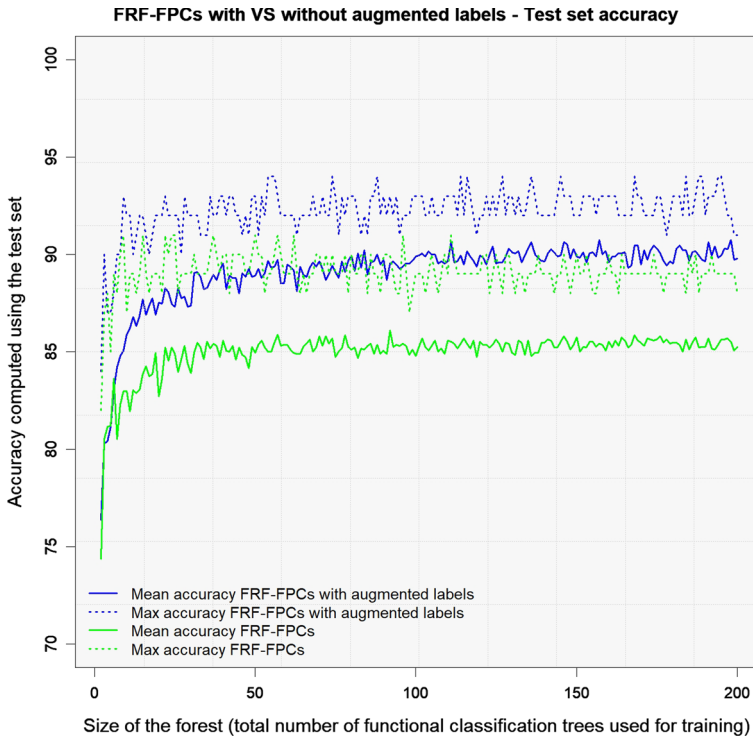


Fig. 11 Comparison of mean and max accuracy computed on the test set between the classical FRF-FPCs and FRF-FPCs with augmented labels classifiers (ECG200 dataset). The mean accuracy is computed using a different number of FPCs (from 2 to 20) given the size of the forest

Despite this last consideration, the result of the FRF-FPCs is fascinating because the average accuracy using the C & T strategy is always much higher than the classical method (solid blue curve in Fig. 16b).

In scenario 4, the C & T procedure turns out to be broadly superior to the classical functional classifiers. Notably, the FRF-B-splines and FRF-FPCs offer compelling results (Fig. 17a, b and c). The accuracy of the FRF-B-spline is systematically higher than that of the classical methods and, therefore, it certainly cannot be due to chance. The same goes for the FRF-FPCs when examining the average accuracies (solid blue curve in Fig. 17b).

Figure 18a, b and c demonstrate that, in the three-classes classification, the C & T process is definitely foremost to classical strategies. Using the FRF-FPCs, the accuracy in Fig. 18b is systematically higher for both the maximum and average versions, and the same is for the FRF-B-splines. Figure 18c depicts that the maximum accuracy is 80% when operating the FKNN, but this result is also achieved without the C & T approach with 3-NN.

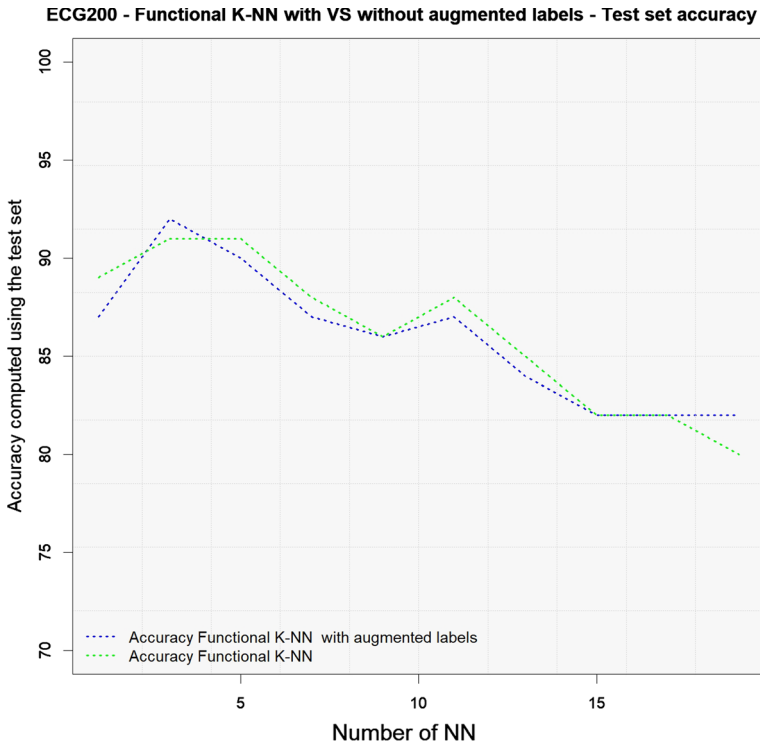


Fig. 12 Test set accuracy comparison between the classical FKM and a FKM with augmented labels (ECG200 dataset)

Scenario 6 deals with the four-classes classification problem. Figures 19b indicates that the FRF-FPCs provides equivalent results both with and without the C & T procedure. Rather, Fig. 19a and c highlight that the C & T technique is still outstanding when applied to the FRF-B-splines and the FKNN.

The details of each simulated scenario, in terms of functional centroids of the identified subgroups, for each original group, are provided in the supplementary material.

4 Discussion and conclusions

In real life, phenomena are frequently characterised by subpatterns, even if concisely categorized into a single class. Therefore, curves with the same class label can have distinct typical behaviours over time. Creating a functional classifier omitting this knowledge is certainly a waste of details that can be used profitably to train a functional classifier. Starting from the high-dimensional data

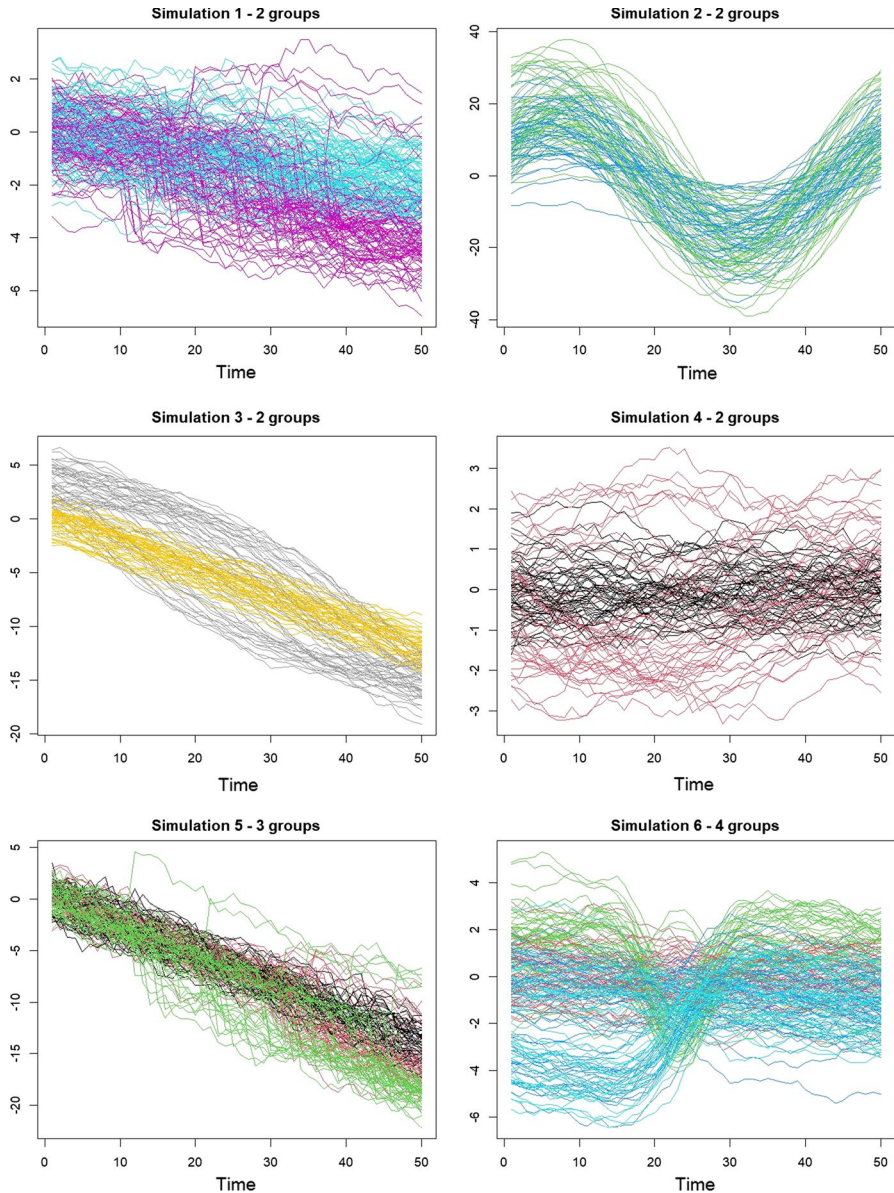


Fig. 13 Simulated scenarios of functional data with two, three, and four classes to predict (more details in the supplementary materials)

classification issue, this work focuses on combining supervised and unsupervised classification in the context of FDA. This research seeks to offer a strategy capable of apprehending additional information on the functional patterns of the

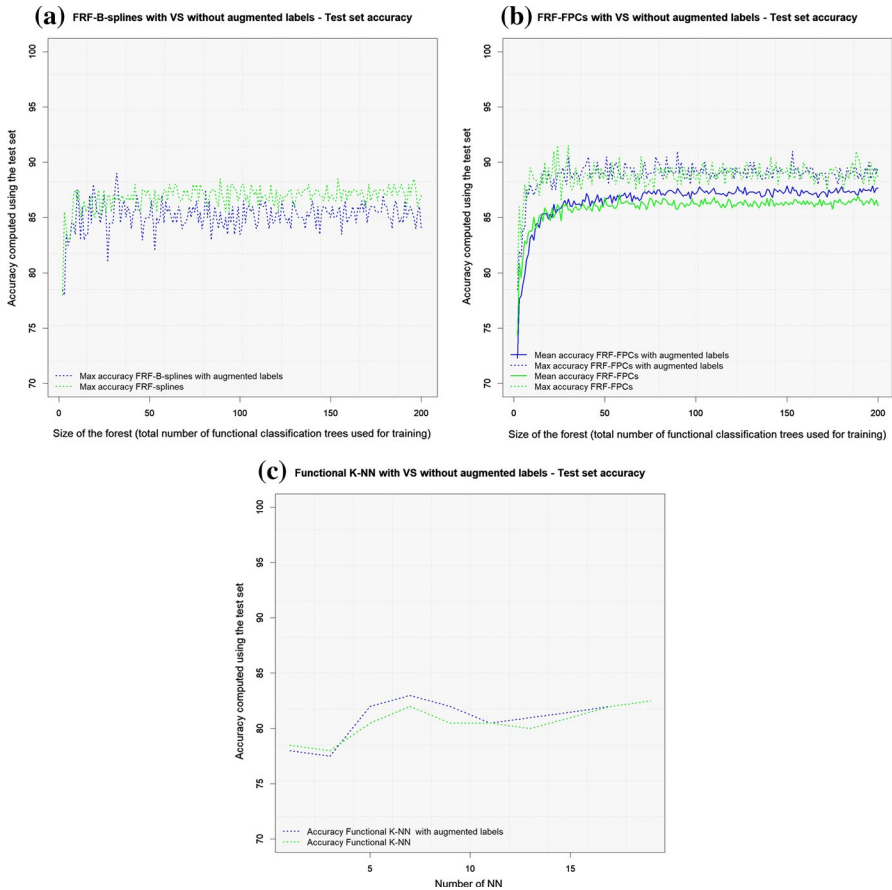


Fig. 14 Scenario 1: **a** test set accuracy comparison between the classical FRF-B-spline and FRF-B-spline using the C &T approach. **b** test set mean and max accuracy comparison between the classical FRF-FPCs and FRF-FPCs using the C &T approach. **c** test set accuracy comparison between the classical FKM and FKM using the C &T approach

original groups of curves to build a more performing functional classifier. The proposed procedure enhances the importance of considering subpatterns in the structure of the functions related to a different behaviour over the time of the observed phenomenon. For this purpose, a two-step method called *Clustering and train* (C &T) is proposed, using first the FKM combined with the FKNN and then the FKM combined with the FRF.

In the first step, a functional clustering algorithm is used to discover new patterns in the original classes. Naturally, it is possible to choose different clustering methods and various metrics or semi-metrics to compute the distance between

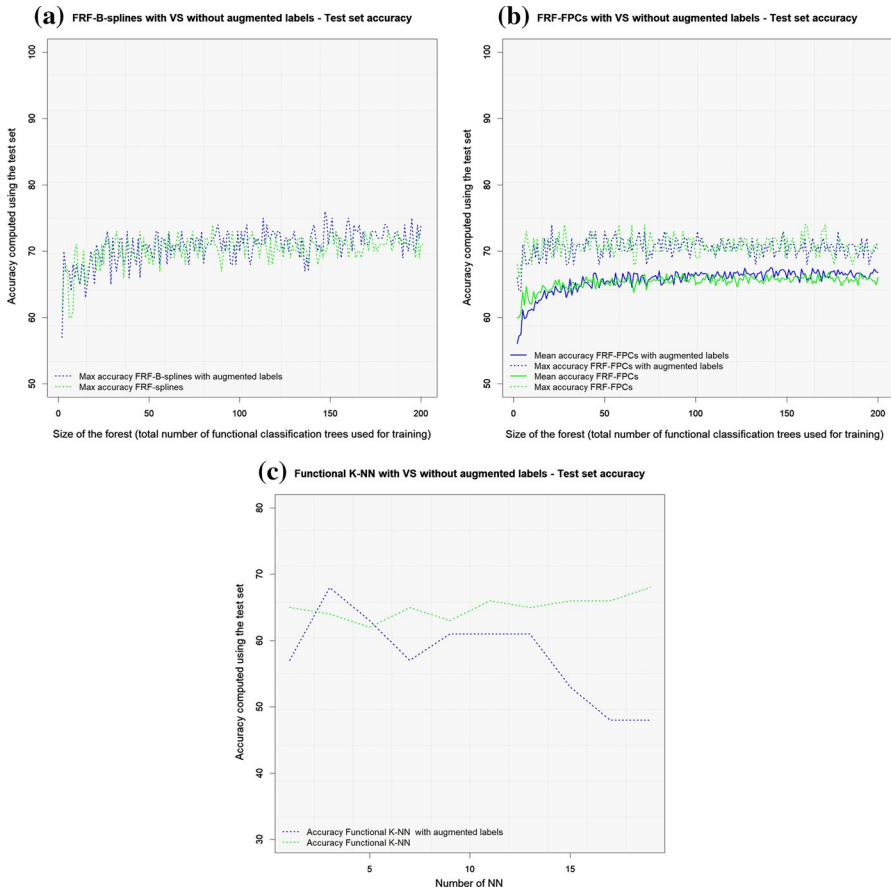


Fig. 15 Scenario 2: **a** test set accuracy comparison between the classical FRF-B-spline and FRF-B-spline using the C & T approach. **b** test set mean and max accuracy comparison between the classical FRF-FPCs and FRF-FPCs using the C & T approach. **c** test set accuracy comparison between the classical FKM and FKM using the C & T approach

curves. At the same time, several strategies can be used to determine the optimal number of subgroups. These options can influence the final results, but are of secondary importance at this stage. Undoubtedly, future studies could explore how different metrics and cluster methods can affect the final result. Nonetheless, in this study, we are interested in obtaining additional knowledge on the existence of functional subgroups of curves belonging to the same initial classes of the outcome to understand if the strategy improves the classic functional classifiers' performance.

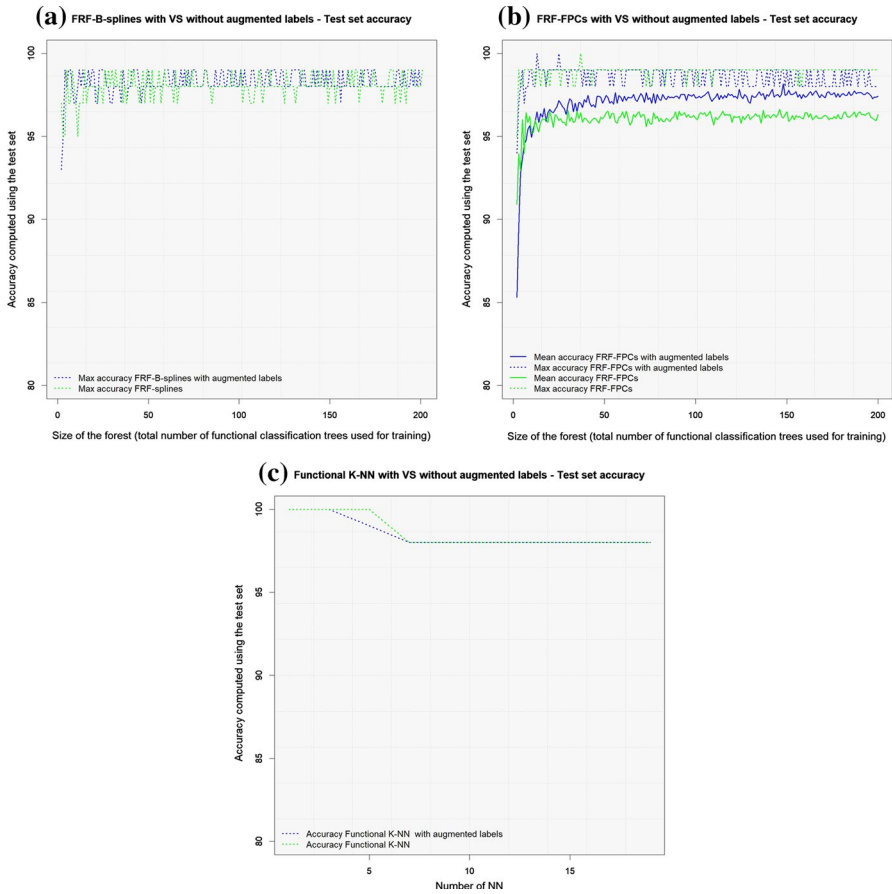


Fig. 16 Scenario 3: **a** test set accuracy comparison between the classical FRF-B-spline and FRF-B-spline using the C & T approach. **b** test set mean and max accuracy comparison between the classical FRF-FPCs and FRF-FPCs using the C & T approach. **c** test set accuracy comparison between the classical FKM and FKM using the C & T approach

This method concentrates on the FKNN and FRF as supervised approaches in the second step. Nevertheless, future investigations may focus on other classifiers to comprehend how the procedure acts with different strategies. The suitable number of subgroups is based on an extension of the silhouette technique to the functional framework through B-splines or FPCs scores. Future studies could focus on looking for methods that support finding the optimal number of subgroups of curves without concentrating on the scores.

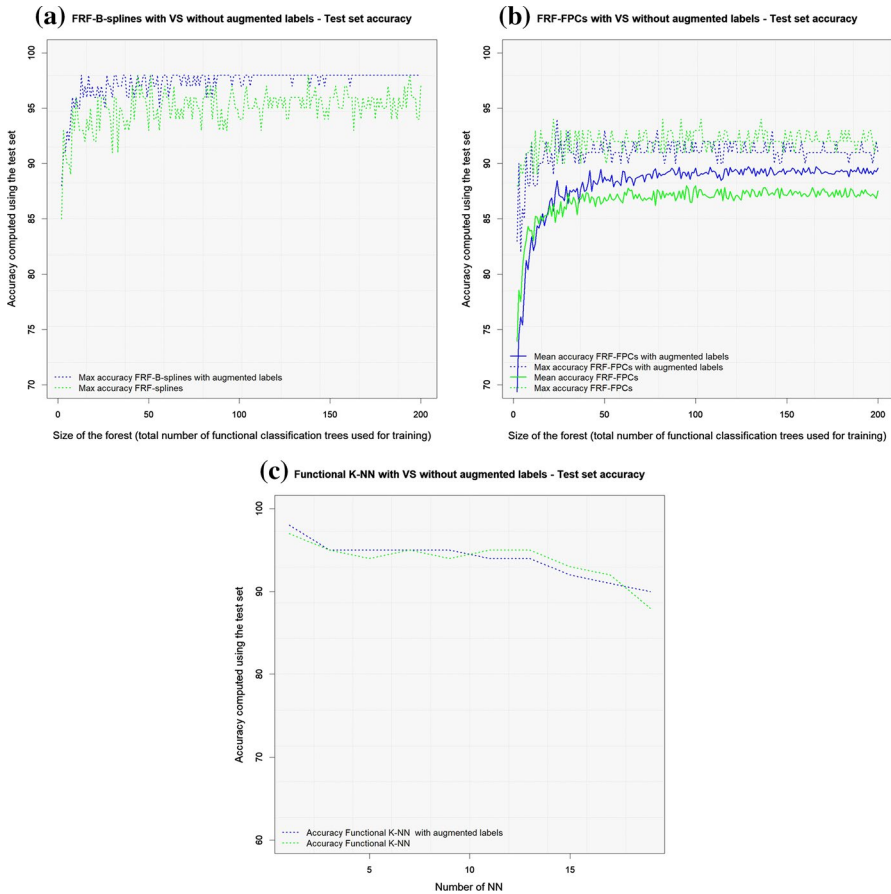


Fig. 17 Scenario 4: **a** test set accuracy comparison between the classical FRF-B-spline and FRF-B-spline using the C & T approach. **b** test set mean and max accuracy comparison between the classical FRF-FPCs and FRF-FPCs using the C & T approach. **c** test set accuracy comparison between the classical FKM and FKM using the C & T approach

The functional two-step procedure, which executes clustering before training, proved to be a reliable tool for catching helpful knowledge on the heterogeneity of the original groups before shifting on to the supervised phase. Definitely, the applications to ECG data and simulated datasets under different scenarios show that the suggested strategy repeatedly leads to a compelling refinement of the functional classifiers' accuracy.

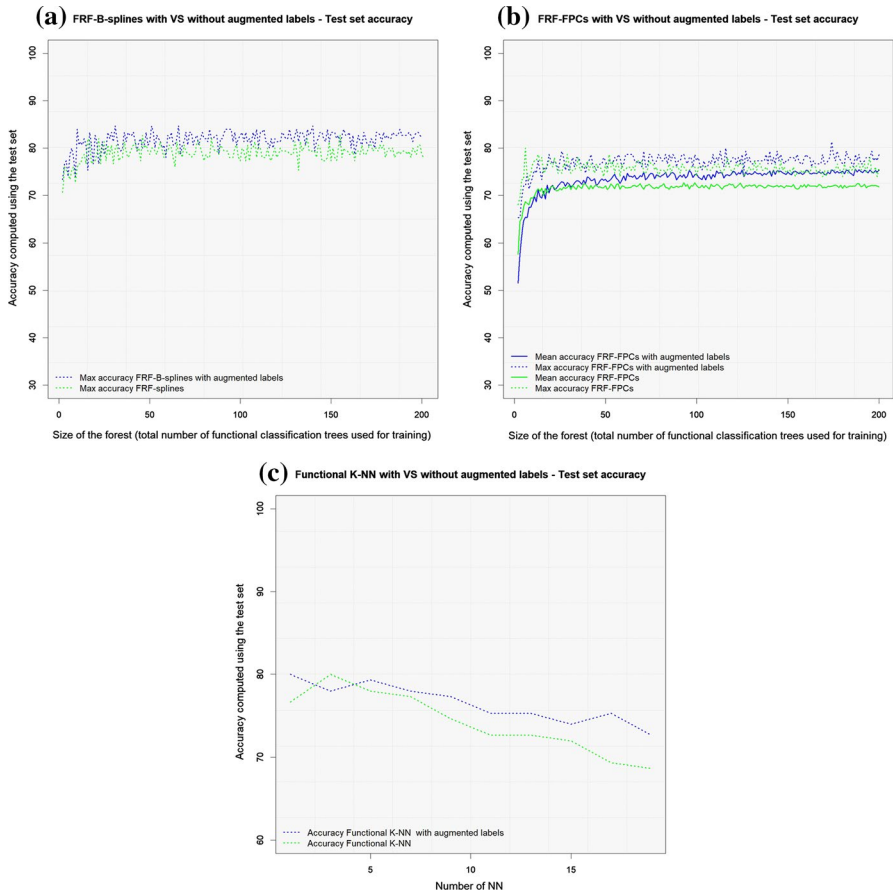


Fig. 18 Scenario 5: **a** test set accuracy comparison between the classical FRF-B-spline and FRF-B-spline using the C &T approach. **b** test set mean and max accuracy comparison between the classical FRF-FPCs and FRF-FPCs using the C &T approach. **c** test set accuracy comparison between the classical FKM and FKM using the C &T approach

Although many possible facets of the proposed procedure can enhance the classifier’s performance, this research delivers fascinating results that could drive further research.

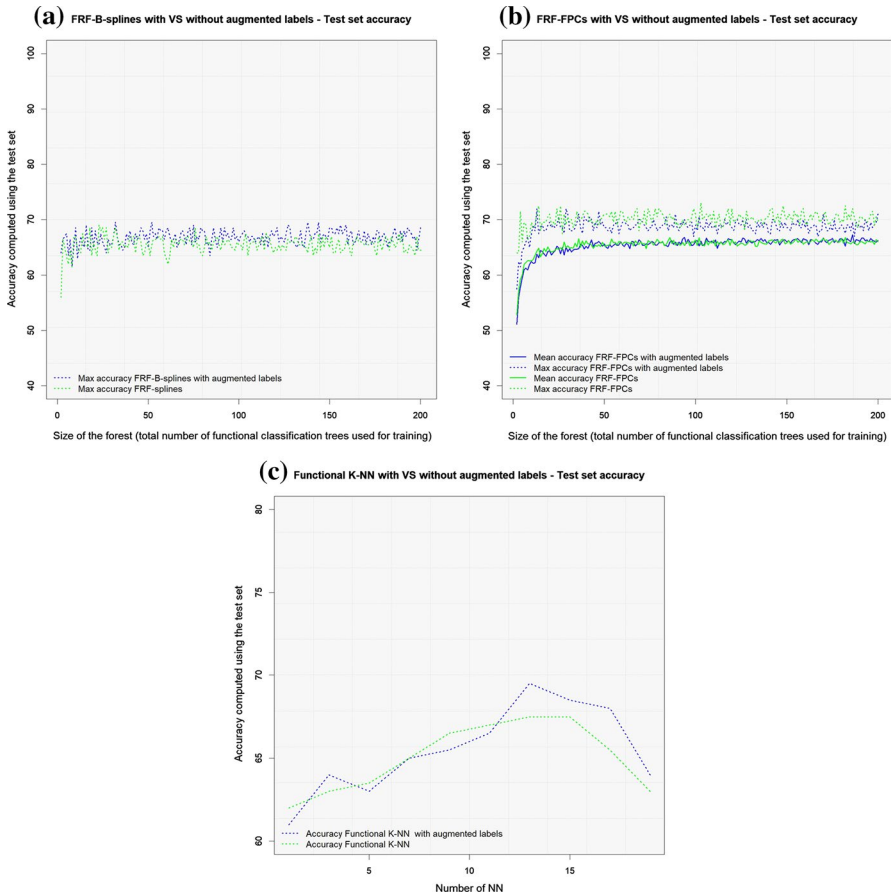


Fig. 19 Scenario 6: **a** test set accuracy comparison between the classical FRF-B-spline and FRF-B-spline using the C & T approach. **b** test set mean and max accuracy comparison between the classical FRF-FPCs and FRF-FPCs using the C & T approach. **c** test set accuracy comparison between the classical FKM and FKM using the C & T approach

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1007/s00180-022-01259-8>.

Funding Open access funding provided by Università degli Studi della Campania Luigi Vanvitelli within the CRUI-CARE Agreement.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Aguilera A, Aguilera-Morillo M (2013) Penalized PCA approaches for B-spline expansions of smooth functional data. *Appl Math Comput* 219:7805–7819. <https://doi.org/10.1016/j.amc.2013.02.009>
- Bagnall A, Lines J, Vickers W, Keogh E (2021) The UEA & UCR time series classification repository. www.timeseriesclassification.com
- Baíllo A, Cuevas A (2008) Supervised classification for functional data: a theoretical remark and some numerical comparisons. *Functional and operatorial statistics*. Physica-Verlag HD, Heidelberg, pp 43–46
- Baíllo A, Cuevas A, Fraiman R (2018) Classification methods for functional data. *Oxford Handbooks Online*
- Breiman L (2004) Random forests. *Mach Learn* 45:5–32. <https://doi.org/10.1023/A:1010933404324>
- Chang C, Chen Y, Ogden RT (2014) Functional data classification: a wavelet approach. *Comput Stat* 29:1497–1513. <https://doi.org/10.1007/s00180-014-0503-4>
- Cuevas A (2014) A partial overview of the theory of statistics with functional data. *J Stat Plan Inference* 147:1–23. <https://doi.org/10.1016/j.jspi.2013.04.002>
- Cuevas A, Febrero M, Fraiman R (2007) Robust estimation and classification for functional data via projection-based depth notions. *Comput Stat* 22(3):481–496. <https://doi.org/10.1007/s00180-007-0053-0>
- Febrero-Bande M, de la Fuente MO (2012) Statistical computing in functional data analysis: the R package *fda.usc*. *J Stat Softw* 51:1–28
- Ferraty F, Vieu P (2003) Curves discrimination: a nonparametric functional approach. *Comput Stat Data Anal* 44(1–2):161–173. [https://doi.org/10.1016/s0167-9473\(03\)00032-x](https://doi.org/10.1016/s0167-9473(03)00032-x)
- Ferraty F, Vieu P (2006) *Nonparametric functional data analysis*. Springer, New York. <https://doi.org/10.1007/0-387-36620-2>
- Fortuna F, Maturo F, Di Battista T (2018) Clustering functional data streams: unsupervised classification of soccer top players based on google trends. *Qual Reliab Eng Int* 34(7):1448–1460. <https://doi.org/10.1002/qre.2333>
- García MLL, García-Ródenas R, Gómez AG (2015) K-means algorithms for functional data. *Neurocomputing* 151:231–245. <https://doi.org/10.1016/j.neucom.2014.09.048>
- Hastie T, Tibshirani R, Friedman J (2009) *The elements of statistical learning: data mining inference and prediction*. Springer, New York. <https://doi.org/10.1007/978-0-387-84858-7>
- Jacques J, Preda C (2013) Functional data clustering: a survey. *Adv Data Anal Classif* 8(3):231–255. <https://doi.org/10.1007/s11634-013-0158-y>
- James G, Witten D, Hastie T, Tibshirani R (2013) *An introduction to statistical learning: with applications in R*. Springer, New York. https://doi.org/10.1007/978-1-4614-7138-7_1
- Maturo F, Verde R (2022) Pooling random forest and functional data analysis for biomedical signals supervised classification: theory and application to electrocardiogram data. *Stat Med* 41:2247–2275. <https://doi.org/10.1002/sim.9353>
- Maturo F, Verde R (2022) Supervised classification of curves via a combined use of functional data analysis and tree-based methods. *Comput Stat TBA*. <https://doi.org/10.1007/s00180-022-01236-1>
- Maturo F, Migliori S, Paolone F (2019) Measuring and monitoring diversity in organizations through functional instruments with an application to ethnic workforce diversity of the U.S. Federal Agencies. *Comput Math Organ Theory* 25(4):357–388. <https://doi.org/10.1007/s10588-018-9267-7>
- Maturo F, Ferguson J, Di Battista T, Ventre V (2020) A fuzzy functional k-means approach for monitoring Italian regions according to health evolution over time. *Soft Comput* 24:13741–13755. <https://doi.org/10.1007/s00500-019-04505-2>
- Milborrow S (2021) *rpart.plot: Plot 'rpart' Models: an enhanced version of 'plot.rpart'*. R package version 3.1.0 <https://CRAN.R-project.org/package=rpart.plot>,
- Mousavi SN, Sørensen H (2018) Functional logistic regression: a comparison of three methods. *J Stat Comput Simul* 88:250–268. <https://doi.org/10.1080/00949655.2017.1386664>
- Olszewski R (2001) Generalized feature extraction for structural pattern recognition in time-series data. *Carnegie-Mellon University, Pittsburgh*
- Preda C, Saporta G, Lévéder C (2007) PLS classification of functional data. *Comput Stat* 22(2):223–235. <https://doi.org/10.1007/s00180-007-0041-4>

- Ramsay J, Silverman B (2005) *Functional Data Analysis*, 2nd edn. Springer, New York. <https://doi.org/10.1007/b98888>
- Ramsay J, Hooker G, Graves S (2009) Introduction to functional data analysis. In: *Functional data analysis with R and MATLAB*, Springer New York, pp 1–19, https://doi.org/10.1007/978-0-387-98185-7_1
- Ramsay JO, Graves S, Hooker G (2022) *fda: Functional data analysis.*, R package version 6.0.3 <https://CRAN.R-project.org/package=fda>
- Taiwo Ojo O, Lillo R, Fernandez Anta A (2021) *fdaoutlier: Outlier detection tools for functional data analysis.* R package version 0.2.0 <https://CRAN.R-project.org/package=fdaoutlier>,
- Therneau T, Atkinson B (2019) *rpart: Recursive partitioning and regression trees.*, R package version 4.1-15 <https://CRAN.R-project.org/package=rpart>
- Yu Y, Lambert D (1999) Fitting trees to functional data, with an application to time-of-day patterns. *J Comput Graph Stat* 8(4):749–762. <https://doi.org/10.1080/10618600.1999.10474847>

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