Statistical Papers (2024) 65:795–825
https://doi.org/10.1007/s00362-023-01408-1

REGULAR ARTICLE

Sparse and smooth functional data clustering

Fabio Centofanti[1] · **Antonio Lepore**[1] · **Biagio Palumbo**[1]

Received: 23 October 2022 / Revised: 8 January 2023 / Published online: 9 March 2023

Abstract
A new model-based procedure is developed for sparse clustering of functional data that aims to classify a sample of curves into homogeneous groups while jointly detecting the most informative portions of the domain. The proposed method is referred to as sparse and smooth functional clustering (SaS-Funclust) and relies on a general functional Gaussian mixture model whose parameters are estimated by maximizing a log-likelihood function penalized with a functional adaptive pairwise fusion penalty and a roughness penalty. The former allows identifying the noninformative portion of the domain by shrinking the means of separated clusters to some common values, whereas the latter improves the interpretability by imposing some degree of smoothing to the estimated cluster means. The model is estimated via an expectation-conditional maximization algorithm paired with a cross-validation procedure. Through a Monte Carlo simulation study, the SaS-Funclust method is shown to outperform other methods that already appeared in the literature, both in terms of clustering performance and interpretability. Finally, three real-data examples are presented to demonstrate the favourable performance of the proposed method. The SaS-Funclust method is implemented in the R package sasfunclust, available on CRAN.

Keywords Functional data analysis · Functional clustering · Model-based clustering · Penalized likelihood · Sparse clustering

✉ Antonio Lepore
antonio.lepore@unina.it

Fabio Centofanti
fabio.centofanti@unina.it

Biagio Palumbo
biagio.palumbo@unina.it

[1] Department of Industrial Engineering, University of Naples Federico II, Piazzale Tecchio 80, 80125 Naples, Italy

1 Introduction

In the last years, due to advances in technology and computational power, most of the data collected by practitioners and scientists in many fields bring information about curves or surfaces that are apt to be modelled as functional data, i.e., continuous random functions defined on a compact domain. A thorough overview of functional data analysis (FDA) techniques can be found in Ramsay and Silverman (2005), Ramsay et al. (2009), Horv'ath and Kokoszka (2012), Hsing and Eubank (2015) and Kokoszka and Reimherr (2017). As in the classical (non-functional) statistical literature, cluster analysis is an important topic in FDA, with many applications in various fields. The primary concern of functional clustering techniques is to classify a sample of data into homogeneous groups of curves, without having any prior knowledge about the true underlying clustering structure. The clustering of functional data is generally a difficult task because of the infinite dimensionality of the problem. For this reason, methods for functional data clustering have received a lot of attention in recent years, and different approaches have been proposed and discussed in the last decade. To the best of the authors' knowledge, the most used approach is the filtering approach (Jacques and Preda 2014), which relies on the reduction of the infinite dimensional problem by approximating functional data in a finite dimensional space and, then, uses traditional clustering tools on the basis expansion coefficients. Along this line, Abraham et al. (2003) propose an advanced version of the k-means algorithm to the coefficients obtained by projecting the original profiles onto a lower-dimensional subspace spanned by B-spline basis functions. A similar method is proposed by Rossi et al. (2004) who apply a Self-Organizing Map (SOM) on the resulting coefficient instead of the k-means algorithm. Elaborating on this path, Serban and Wasserman (2005) present a technique for the nonparametric estimation and clustering of a large number of functional data that is still based on the k-means algorithm applied to the basis expansion coefficients obtained through smoothing techniques. A step forward is moved by Chiou and Li (2007), who introduce the k-centers functional clustering method to account, differently from the previous methods, for both the means and the mode of variation differentials between clusters by predicting cluster membership with a reclassification step.

Instead of considering the basis expansion coefficients as parameters, a different idea is that of using a model-based approach where coefficients are treated as random variables themselves with a cluster-specific probability distribution. The seminal work of James and Sugar (2003) is the first one to develop a flexible model-based procedure to cluster functional data based on a random effects model for the coefficients. This allows for borrowing strength across curves and, thus, for superior results when data contain a large number of sparsely sampled curves. More recently, Bouveyron and Jacques (2011) propose a new functional clustering method to fit the functional data in group-specific functional subspaces, which is referred to as funHDDC and is based on a functional latent Gaussian mixture model. By constraining model parameters within and between groups, they obtain a family of parsimonious models that allow for more flexibility. Analogously, Jacques and Preda (2013) assume cluster-specific Gaussian distribution on the principal components resulting from the Karhunen-Loeve expansion of the curves, and Giacofci et al. (2013) propose to use a Gaussian mixture

model on the wavelet decomposition of the curves, which turns out to be particularly appropriate for peak-like data, as opposed to methods based on splines.

In the multivariate cluster analysis, some attributes could be, however, completely noninformative for uncovering the clustering structure of interest. As an example, this often happens in high-dimensional problems, i.e., where the number of variables is larger than the number of observations. In this setting, the task of identifying the features, in which respect true clusters differ the most, is of great interest to achieve a more accurate identification of the groups, as noninformative features may hide the true clustering structure, and higher interpretability of the analysis, by imputing the presence of the clustering structure to a small number of features. More in general, the methods capable of selecting informative features and eliminating noninformative ones are referred to as *sparse*. Such a class of methods can be usually reconducted and regarded as variable selection methods. Sparse clustering has received increasing attention in the recent literature. Based on conventional heuristic clustering algorithms, Friedman and Meulman (2004) develop a new procedure to automatically detect subgroups of objects, which preferentially cluster on subsets of features. Witten and Tibshirani (2010) elaborate a novel clustering framework based on an adaptively chosen subset of features that are selected by means of a lasso-type penalty. In terms of model-based approaches, the method introduced by Raftery and Dean (2006) is able to sequentially compare nested models through the approximate Bayes factor and to select the informative features. Maugis et al. (2009) improve this method by considering the noninformative features as independent from some informative ones.

It is moreover worth mentioning quite promising variable selection approaches that make use of a regularization framework. The seminal work in this direction is that of Pan and Shen (2007), who introduce a penalized likelihood approach with an L_1 penalty function, which is able to automatically achieve variable selection via thresholding and delivering a sparse solution. Similarly, Wang and Zhu (2008) suggest a solution by replacing the L_1 penalty with either the L_∞ penalty or the hierarchical penalization function, which takes into account the fact that cluster means, corresponding to the same feature, can be treated as grouped. Xie et al. (2008) also account for grouped parameters through the use of two planes of grouping, named vertical and horizontal grouping. In all sparse clustering methods just mentioned, a feature is selected if it is informative for at least one pair of clusters and eliminated otherwise, i.e., if it is noninformative for all clusters. However, some variables could be informative only for specific pairs of clusters. For this reason, Guo et al. (2010) propose a pairwise fusion penalty that penalizes, for each feature, the differences between all pairs of cluster means and fuses only the non-separated clusters.

Only recently, the notion of sparseness has been translated into a functional data clustering framework. Specifically, sparse functional clustering methods aim to cluster the curves while jointly detecting the most informative portion of the domain to the clustering in order to improve both the accuracy and the interpretability of the analysis. Based on the idea of Chen et al. (2014), Floriello and Vitelli (2017) propose a sparse functional clustering method based on the estimation of a suitable weight function that is capable of identifying the informative part of the domain. Vitelli (2019) proposes a novel framework for sparse functional clustering that also embeds an alignment step. Moreover, Cremona and Chiaromonte (2022) develop a new method to locally cluster

curves and discover functional motifs, and Di Iorio and Vantini (2019) introduce *fun*BI, the first biclustering algorithm for functional data. To the best of the authors' knowledge, these are the only works that propose sparse functional clustering methods so far.

In this article, we present a model-based procedure for the sparse clustering of functional data, named sparse and smooth functional clustering (SaS-Funclust), where the basic idea is to provide accurate and interpretable cluster analysis. Specifically, the parameters of a general functional Gaussian mixture model are estimated by maximizing a penalized version of the log-likelihood function, where a functional adaptive pairwise fusion penalty, the functional extension of the penalty proposed by Guo et al. (2010), is introduced. The latter penalizes the pointwise differences between all pairs of cluster functional means and locally shrinks the means of cluster pairs to some common values. Then, a roughness penalty on cluster functional means is considered to further improve the interpretability of the cluster analysis. By this, the SaS-Funclust method gains the ability to detect, for each cluster pair, the informative portion of the domain to the clustering, hereinafter always intended in terms of mean differences. If a specific mean pair is fused over a portion of the domain, it is labelled as noninformative to the clustering of that pair. Otherwise, it is labelled as informative. In other words, the proposed method is able to detect portions of the domain that are noninformative *pairwise*, i.e., at least for a specific cluster pair, differently from the method proposed by Floriello and Vitelli (2017) that is only able to detect portions of the domain that are noninformative *overall*, i.e., for all the cluster pairs simultaneously. Moreover, the model-based fashion of the proposed method provides greater flexibility than the latter, which basically relies on k-means clustering. A specific expectation-conditional maximization (ECM) algorithm is designed to perform the maximization of the penalized log-likelihood function, which is a non-trivial problem, and a cross-validation based procedure is proposed to select the appropriate model.

To give a general idea of the pairwise sparseness property of the proposed method, Fig. 1shows the cluster means estimated by the latter for three different simulated data sets with (a) two, (b) three, and (c) four clusters. Data are generated as described in Sect. 3 and supplementary information S2.

In Fig. 1a, the estimated means are correctly fused over $t \in (0.2, 1.0]$. Hence, the proposed method is shown to be able to identify the informative portion of domain $[0.0, 0.2]$, for the unique pair of clusters and not for all. In Fig. 1b and c, several cluster pairs are available, because the number of clusters is larger than 2, and, thus, a given portion of the domain could be informative for a specific pair of clusters. In Fig. 1b, the informative portion of the domain for each pair of clusters is correctly recovered. The estimated cluster means are indeed pairwise fused over approximately the same portion of the domain as the true cluster means pairs. Note that, for the clusters whose true means are equal over $t \in (0.2, 1.0]$, the SaS-Funclust method identifies the informative portion of the domain roughly in $[0.0, 0.2]$. In Fig. 1c, the sparseness property of the SaS-Funclust method is even more striking. In this case, in the face of many cluster pairs, the proposed method is still able to successfully detect the informative portion of the domain.

The innovation and advantage of SaS-Funclust over already existing methods can be synopsized as follows. With respect to the multivariate literature, the SaS-Funclust

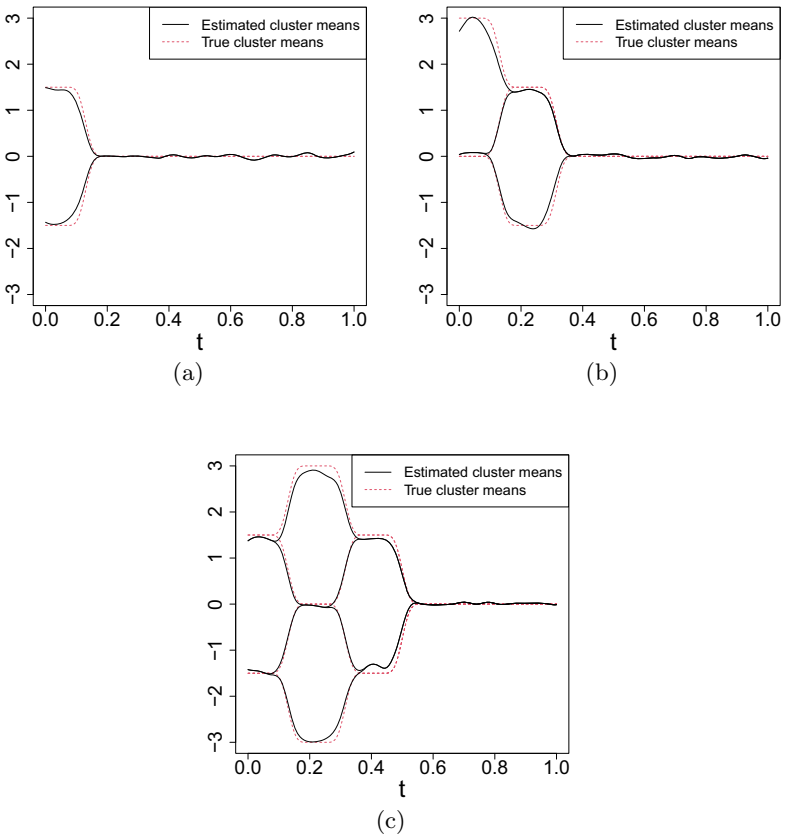

Fig. 1 True and estimated cluster means obtained through the SaS-Funclust method for three different simulated data sets with **a** two, **b** three and **c** four clusters generated as described in Sect. 3

method is able to extend the advantages of sparsity, i.e., the capability of selecting informative features and eliminating noninformative ones, to the functional data setting and achieving larger accuracy in the identification of the groups, as noninformative features may hide the true clustering structure, and interpretability of the results, which has the potential of improving the degree of understanding of the process under study. With respect to the *sparse* and *functional* clustering methods already presented in the literature before, SaS-Funclust is the first *model-based* approach and is thus expected to attain superior flexibility in modelling different cluster shapes. Moreover, these competing methods are only able to detect portions of the domain that are noninformative *overall*, whereas, to the best of the authors' knowledge, SaS-Funclust is able to more efficiently detect informative portions of the domain in a pairwise fashion, as depicted in Fig. 1.

The remainder of this article is organized as follows. After the presentation of the proposed method in Sect. 2, its finite-sample properties will be addressed in Sect. 3 through a wide Monte Carlo simulation where we further demonstrate its favourable performance, both in terms of clustering accuracy and interoperability, over several

competing methods. In Sect. 4, the application of the proposed method to three real-data examples, i.e., the Berkeley Growth Study, the Canadian weather, and the ICOSAF project data, remark the practical advantages and potentiality of the proposed method that proves to attain, thanks to its sparseness property, new insightful and interpretable solutions to cluster analysis. Section 5 concludes the paper. The method presented in this article is implemented in the R package sasfunclust, openly available on CRAN.

2 The SaS-Funclust method for functional clustering

In this section, we present the key elements of the proposed method. Specifically, Sects. 2.1 and 2.2 introduce the general functional Gaussian mixture model and the penalized maximum likelihood estimator, respectively. Whereas, the optimization algorithm and parameter selection considerations are discussed in Sects. 2.3 and 2.4, respectively.

2.1 A general functional clustering model

The SaS-Funclust method is based on the general functional clustering model introduced by James and Sugar (2003). Suppose that N observations are spread among G unknown clusters and that the probability of each observation belonging to the gth cluster is π_g, $\sum_{g=1}^{G} \pi_g = 1$. Moreover, let us denote with $\boldsymbol{Z}_i = (Z_{1i}, \ldots, Z_{Gi})^T$ the unknown component-label vector corresponding to the ith observation, where Z_{gi} equals 1 if the ith observation is in the gth cluster and 0 otherwise. Then, let us assume that for each i observation, $i = 1, \ldots, N$ in the cluster $g = 1, \ldots, G$, it is available a vector $\boldsymbol{Y}_i = \left(y_{i1}, \ldots, y_{in_i}\right)^T$ of size n_i, which can differ across observations, of observed values of a function g_i over the time points $t_{i1}, \ldots, t_{in_i}$. The function g_i is assumed a Gaussian random process with mean μ_g, covariance ω_g, and values in $L^2(\mathcal{T})$, the separable Hilbert space of square integrable functions defined on the compact domain $\mathcal{T}$. We assume that, conditionally on $Z_{gi} = 1$, $\boldsymbol{Y}_i$ is modelled as

$$\boldsymbol{Y}_i = \boldsymbol{g}_i + \boldsymbol{\epsilon}_i, \quad i = 1, \ldots, N, \tag{1}$$

where $\boldsymbol{g}_i = \left(g_i(t_{i1}), \ldots, g_i\left(t_{in_i}\right)\right)^T$ contains the values of the function g_i at $t_{i1}, \ldots, t_{in_i}$ and $\boldsymbol{\epsilon}_i$ is a vector of measurement errors that are mutually independent and normally distributed with zero mean and constant variance σ_e^2. Let us suppose also that the unknown component-label vector $\boldsymbol{Z}_i$ has a multinomial distribution, which consists of one draw on g categories with probabilities $\pi_1, \ldots, \pi_G$. Then, for every i, the unconditional density function $f(\cdot)$ of $\boldsymbol{Y}_i$ is

$$f(\boldsymbol{Y}_i) = \sum_{g=1}^{G} \pi_g \psi\left(\boldsymbol{Y}_i; \boldsymbol{\mu}_{gi}, \boldsymbol{\Omega}_{gi} + \boldsymbol{I}\sigma_e^2\right), \tag{2}$$

where $\boldsymbol{\mu}_{gi} = \left(\mu_g\left(t_{i1}\right), \ldots, \mu_g\left(t_{in_i}\right)\right)^T$, $\boldsymbol{\Omega}_{gi} = \{\omega_g\left(t_{ki}, t_{li}\right)\}_{k,l=1\ldots,n_i}$, $\boldsymbol{I}$ is the identity matrix, and $\psi\left(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$ is the multivariate Gaussian density distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. The model in Eq. (2) is the classical G-component Gaussian mixture model (McLachlan and Peel 2004).

As discussed in James and Sugar (2003), it is necessary to impose some structure curves g_i, because both curves could be observed at different time domain points and the dimensionality of the model in Eq. (2) could be too high in comparison to the sample size. Therefore, similarly to the filtering approach for clustering (Capezza et al. 2021), we assume that each function g_i, for $i = 1, \ldots, N$, may be represented in terms of a q-dimensional set of basis functions $\boldsymbol{\Phi} = \left(\phi_1, \ldots, \phi_q\right)^T$, that is

$$g_i\left(t\right) = \boldsymbol{\eta}_i^T \boldsymbol{\Phi}\left(t\right), \quad t \in \mathcal{T}, \tag{3}$$

where $\boldsymbol{\eta}_i = \left(\eta_{i1}, \ldots, \eta_{iq}\right)^T$ are vectors of basis coefficients. Then, $\boldsymbol{\eta}_i$ are modelled as Gaussian random vectors, that is, given that $Z_{gi} = 1$,

$$\boldsymbol{\eta}_i = \boldsymbol{\mu}_g + \boldsymbol{\gamma}_{ig}, \tag{4}$$

where $\boldsymbol{\mu}_g = \left(\mu_{g1}, \ldots, \mu_{gq}\right)^T$ are q-dimensional vectors and $\boldsymbol{\gamma}_{ig}$ are Gaussian random vectors with zero mean and covariance $\boldsymbol{\Gamma}_g$. With these assumption, the unconditional density function $f\left(\cdot\right)$ of $\boldsymbol{Y}_i$ in Eq. (2) becomes

$$f\left(\boldsymbol{Y}_i\right) = \sum_{g=1}^{G} \pi_g \psi\left(\boldsymbol{Y}_i; \boldsymbol{S}_i\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_{ig}\right), \tag{5}$$

where $\boldsymbol{S}_i = \left(\boldsymbol{\Phi}\left(t_{i1}\right), \ldots, \boldsymbol{\Phi}\left(t_{in_i}\right)\right)^T$ is the basis matrix for the ith curve and $\boldsymbol{\Sigma}_{ig} = \boldsymbol{S}_i\boldsymbol{\Gamma}_g\boldsymbol{S}_i^T + \boldsymbol{I}\sigma_e^2$. Therefore, the log-likelihood function corresponding to $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_N$ is given by

$$L\left(\boldsymbol{\Theta}|\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_N\right) = \sum_{i=1}^{N} \log \sum_{g=1}^{G} \pi_g \psi\left(\boldsymbol{Y}_i; \boldsymbol{S}_i\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_{ig}\right), \tag{6}$$

where $\boldsymbol{\Theta} = \{\pi_g, \boldsymbol{\mu}_g, \boldsymbol{\Gamma}_g, \sigma_e^2\}_{g=1,\ldots,G}$ is the parameter set of interest. Based on an estimate $\hat{\boldsymbol{\Theta}} = \{\hat{\pi}_g, \hat{\boldsymbol{\mu}}_g, \hat{\boldsymbol{\Gamma}}_g, \hat{\sigma}_e^2\}_{g=1,\ldots,G}$, an observation $\boldsymbol{Y}^*$ is assigned to the cluster g that achieves the largest posterior probability estimate $\hat{\pi}_g \psi\left(\boldsymbol{Y}^*; \boldsymbol{S}_i\hat{\boldsymbol{\mu}}_g, \hat{\boldsymbol{\Sigma}}_{ig}\right)$, with $\hat{\boldsymbol{\Sigma}}_{ig} = \boldsymbol{S}_i\hat{\boldsymbol{\Gamma}}_g\boldsymbol{S}_i^T + \boldsymbol{I}\hat{\sigma}_e^2$.

2.2 The penalized maximum likelihood estimator

James and Sugar (2003) propose to estimate $\boldsymbol{\Theta}$ through the maximum likelihood estimator (MLE), which is the maximizer of the log-likelihood function in Eq. (6). In

this work, we propose a different estimator of $\boldsymbol{\Theta}$ that is the maximizer of the following penalized log-likelihood

$$L_p\left(\boldsymbol{\Theta}|\boldsymbol{Y}_1,\ldots,\boldsymbol{Y}_N\right)=\sum_{i=1}^{N}\log\sum_{g=1}^{G}\pi_g\psi\left(\boldsymbol{Y}_i;\boldsymbol{S}_i\boldsymbol{\mu}_g,\boldsymbol{\Sigma}_{ig}\right)-\mathcal{P}\left(\boldsymbol{\mu}_g\right), \tag{7}$$

where $\mathcal{P}(\cdot)$ is a penalty function defined as

$$\begin{aligned}\mathcal{P}\left(\boldsymbol{\mu}_g\right) &= \lambda_L\sum_{1\le g\le g'\le G}\int_{\mathcal{T}}\tau_{g,g'}(t)\,|\mu_g(t)-\mu_{g'}(t)|dt\\ &+\lambda_s\sum_{g=1}^{G}\int_{\mathcal{T}}\left(\mu_g^{(s)}(t)\right)^2dt,\end{aligned} \tag{8}$$

where $\lambda_L,\lambda_s\ge 0$ are tuning parameters, and $\tau_{g,g'}$ are prespecified weight functions. The symbol $f^{(s)}(\cdot)$ denotes the sth-order derivative of f if it is a function or the entries of f if it is a vector. Note that in Eq. (8) each function g_i may be represented as in Eq. (3), then it follows that

$$\begin{aligned}\mathcal{P}\left(\boldsymbol{\mu}_g\right) &= \lambda_L\sum_{1\le g\le g'\le G}\int_{\mathcal{T}}\tau_{g,g'}(t)\,|\boldsymbol{\mu}_g^T\boldsymbol{\Phi}(t)-\boldsymbol{\mu}_{g'}^T\boldsymbol{\Phi}(t)|dt\\ &+\lambda_s\sum_{g=1}^{G}\int_{\mathcal{T}}\left(\boldsymbol{\mu}_g^T\boldsymbol{\Phi}^{(s)}(t)\right)^2dt,\end{aligned} \tag{9}$$

The first element of the right-hand side of Eq. (8) is the functional extension of the penalty introduced by Guo et al. (2010) and is referred to as functional adaptive pairwise fusion penalty (FAPFP). The aim of the FAPFP is to shrink the differences between every pair of cluster means for each value of $t\in\mathcal{T}$. Due to the property of the absolute value function of being singular at zero, some of these differences are shrunken exactly to zero. In particular, the FAPFP allows pair of cluster means to be equal over a specific portion of the domain that is, thus, considered noninformative for separating the means of that pair of clusters.

The choice of the weight function $\tau_{g,g'}$ in Eqs. (8) and (9) should be based on the idea that if a given portion of the domain is informative for separating the means of the corresponding pair of clusters, then, the values of $\tau_{g,g'}$ over that portion should be small. In this way, the absolute difference $|\mu_g(\cdot)-\mu_{g'}(\cdot)|$ is penalized more over noninformative portions of the domain than over informative ones. Following the standard practice for the adaptive penalties (Zou 2006), we propose to use

$$\tau_{g,g'}(t)=|\tilde{\mu}_g(t)-\tilde{\mu}_{g'}(t)|^{-1}\quad t\in\mathcal{T}, \tag{10}$$

where $\tilde{\mu}_g$ are initial estimates of the cluster means.

Finally, the term $\lambda_s \sum_{g=1}^{G} \int_{\mathcal{T}} \left(\mu_g^{(s)}(t)\right)^2 dt$ is a smoothness penalty that penalizes the sth derivative of the cluster means. This term aims to further improve the interpretability of the results by constraining, with a magnitude quantified by λ_s, the cluster means to own a certain degree of smoothness, measured by the derivative order s. Following the common practice in FDA (Ramsay and Silverman 2005), the natural choice to penalize the cluster mean curvature is to set $s = 2$, which implies that the chosen basis functions are differentiable at least s times. As a remark, the penalization in Eq. (7) is applied only to the cluster mean coefficients $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_G$. The reason is that, as previously stated in the introduction a portion of the domain is defined as informative only in terms of cluster mean differences. However, portions of the domain could be informative also in terms of differences in cluster covariances, which together with the means uniquely identify each cluster.

2.3 The penalty approximation and the optimization algorithm

To perform the maximization of the penalized log-likelihood in Eq. (7), the penalty $\mathcal{P}(\cdot)$, defined as in Eq. (8), can be written as

$$\begin{aligned}\mathcal{P}\left(\boldsymbol{\mu}_g\right) &= \lambda_L \sum_{1 \le g \le g' \le G} \int_{\mathcal{T}} |\left(\tilde{\boldsymbol{\mu}}_g^T - \tilde{\boldsymbol{\mu}}_{g'}\right)^T \boldsymbol{\Phi}(t)|^{-1} |\left(\boldsymbol{\mu}_g^T - \boldsymbol{\mu}_{g'}\right)^T \boldsymbol{\Phi}(t)| dt \\ &+ \lambda_s \sum_{g=1}^{G} \boldsymbol{\mu}_g^T \boldsymbol{W} \boldsymbol{\mu}_g, \end{aligned} \tag{11}$$

where the weight functions $\tau_{g,g'}(t)$ are expressed as in Eq. (10), and the initial estimates of the cluster means are represented through the set of basis functions $\boldsymbol{\Phi}$ as $\tilde{\mu}_g(t) = \tilde{\boldsymbol{\mu}}_{g'}^T \boldsymbol{\Phi}(t)$, $t \in \mathcal{T}$, with $\tilde{\boldsymbol{\mu}}_g = \left(\tilde{\mu}_{g1}, \ldots, \tilde{\mu}_{gq}\right)^T$. The matrix $\boldsymbol{W}$ is equal to $\int_{\mathcal{T}} \boldsymbol{\Phi}^{(s)}(t) \left(\boldsymbol{\Phi}^{(s)}(t)\right)^T dt$. A great simplification of the optimization problem can be achieved if the first element on the right-hand side of Eq. (11) can be expressed as a linear function of $|\boldsymbol{\mu}_g^T - \boldsymbol{\mu}_{g'}|$. The following theorem provides a practical way to rewrite the first term of the right-hand side of Eq. (11) as linear function of $|\boldsymbol{\mu}_g^T - \boldsymbol{\mu}_{g'}|$, when $\boldsymbol{\Phi}$ is a set of B-splines (De Boor et al. 1978; Schumaker 2007).

Theorem 1 *Let $\boldsymbol{\Phi} = \left(\phi_1, \ldots, \phi_q\right)^T$ be the set of B-splines of order k and non-decreasing knots sequences $\{x_0, x_1, \ldots, x_{M_j}, x_{M+1}\}$ defined on the compact set $\mathcal{T} = \left[x_0, x_{M+1}\right]$, with $q = M + k$, and $\{\tau_j\}_{j=1}^{q+1}$ being a sequence with $\tau_1 = x_0$, $\tau_j = \tau_{j-1} + \left(x_{\min(M+1,j)} - x_{\max(0,j-1-k)}\right)/k$, $\tau_{q+1} = x_{M+1}$. Then, for each function $f(t) = \sum_{i=1}^{q} c_i \phi_i(t)$, $t \in \mathcal{T}$, where $c_i \in \mathbb{R}$, the function $\tilde{f}(t) = \sum_{i=1}^{q} c_i I_{[\tau_i, \tau_{i+1}]}(t)$, $t \in \mathcal{T}$, where $I_{[\tau_i, \tau_{i+1}]}(t) = 1$ for $t \in \left[\tau_i, \tau_{i+1}\right]$ and zero elsewhere, is such that*

$$\sup_{t \in \mathcal{T}} |f(t) - \tilde{f}(t)| = O(\delta), \tag{12}$$

where $\delta = \max_i |x_{i+1} - x_i|$*, that is* $f(t) - \tilde{f}(t)$ *converges uniformly to the zero function.*

Theorem 1, whose proof is deferred to the supplementary information S1, basically states that when δ is small, f is well approximated by $\tilde{f}$. In other words, the approximation error $|f - \tilde{f}|$ can be made arbitrarily small by increasing the number of knots. If we further assume the knots sequence is evenly spaced, δ turns out to be equal to $1/M$. These considerations allow us to approximate $|\left(\boldsymbol{\mu}_g^T - \boldsymbol{\mu}_{g'}\right)^T \boldsymbol{\Phi}(t)|$ and $|\left(\tilde{\boldsymbol{\mu}}_g^T - \tilde{\boldsymbol{\mu}}_{g'}\right)^T \boldsymbol{\Phi}(t)|$, respectively, as follows

$$\begin{aligned} |\left(\boldsymbol{\mu}_g - \boldsymbol{\mu}_{g'}\right)^T \boldsymbol{\Phi}(t)| &\approx |\boldsymbol{\mu}_g - \boldsymbol{\mu}_{g'}|^T \boldsymbol{I}(t), \quad \forall t \in \mathcal{T} \\ |\left(\tilde{\boldsymbol{\mu}}_g - \tilde{\boldsymbol{\mu}}_{g'}\right)^T \boldsymbol{\Phi}(t)| &\approx |\tilde{\boldsymbol{\mu}}_g - \tilde{\boldsymbol{\mu}}_{g'}|^T \boldsymbol{I}(t), \quad \forall t \in \mathcal{T} \end{aligned} \tag{13}$$

where $\boldsymbol{I} = \left(I_{[\tau_1,\tau_2]}, \ldots, I_{[\tau_q,\tau_{q+1}]}\right)^T$. Thus, Eq. (11) can be rewritten as

$$\mathcal{P}\left(\boldsymbol{\mu}_g\right) = \lambda_L \sum_{1 \le g \le g' \le G} \tilde{\boldsymbol{m}}^T |\boldsymbol{\mu}_g - \boldsymbol{\mu}_{g'}| + \lambda_s \sum_{g=1}^{G} \boldsymbol{\mu}_g^T \boldsymbol{W} \boldsymbol{\mu}_g, \tag{14}$$

where $\tilde{\boldsymbol{m}} = \left(\frac{\tau_2 - \tau_1}{|\tilde{\mu}_{g1} - \tilde{\mu}_{g'1}|}, \ldots, \frac{\tau_{q+1} - \tau_q}{|\tilde{\mu}_{gq} - \tilde{\mu}_{g'q}|}\right)^T$.

The goodness of the approximations in Eq. (13) depends on the cardinality q of the set of B-splines $\boldsymbol{\Phi}$, which should be as large as possible. However, the number of parameters in Eq. (2), which depends quadratically on q, becomes very large even for moderate values of q. This issue can be mitigated if one may further assume equal and diagonal coefficient covariance matrices across all clusters, that is $\boldsymbol{\Gamma}_1 = \cdots = \boldsymbol{\Gamma}_G = \boldsymbol{\Gamma} = \text{diag}\left(\sigma_1^2, \ldots, \sigma_q^2\right)$. This assumption implies that clusters are separated only by their mean values, which is coherent with the general premise that the informative portion of the domain is identified only by cluster mean differences.

The penalized log-likelihood function in Eq. (7) then becomes

$$\begin{aligned} L_p\left(\boldsymbol{\Theta} | \boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_N\right) = &\sum_{i=1}^{N} \log \sum_{g=1}^{G} \pi_g \psi\left(\boldsymbol{Y}_i; \boldsymbol{S}_i \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_i\right) \\ &- \lambda_L \sum_{1 \le g \le g' \le G} \tilde{\boldsymbol{m}}^T |\boldsymbol{\mu}_g - \boldsymbol{\mu}_{g'}| - \lambda_s \sum_{g=1}^{G} \boldsymbol{\mu}_g^T \boldsymbol{W} \boldsymbol{\mu}_g, \end{aligned} \tag{15}$$

with $\boldsymbol{\Sigma}_i = \boldsymbol{S}_i \boldsymbol{\Gamma} \boldsymbol{S}_i^T + \boldsymbol{I} \sigma_e^2$. Note that, in Eq. (15), the FAPFP is approximated through the sum of weighted linear combinations of the absolute values of the coefficient differences between every pair of cluster means, which strictly resembles the multivariate LASSO penalty applied to the differences of the basis expansion coefficients,

i.e. $\lambda_L \sum_{1\le g\le g'\le G} |\boldsymbol{\mu}_g - \boldsymbol{\mu}_{g'}|$. However, the presence of $\tilde{\boldsymbol{m}}$ in the FAPFP approximation is crucial because it allows the penalty to differently shrink coefficient differences corresponding to B-splines with different support. That is, it avoids coefficient differences, which correspond to B-splines strictly localized, are weighted as coefficient differences of spreader basis in the computation of the approximated penalty. This also means that the presence of $\tilde{\boldsymbol{m}}$ allows the proposed approximation of the FAPFP to achieve a lower error than the multivariate LASSO penalty applied to the coefficient differences.

The maximization of this objective function is a nontrivial problem. A specifically designed algorithm is proposed, which is a modification of the expectation maximization (EM) algorithm proposed by James and Sugar (2003). By treating the component-label vectors $\boldsymbol{Z}_i$ (defined at the beginning of Sect. 2.1) and $\boldsymbol{\gamma}_{ig}$ (see Eq. (4)) as missing data, the complete penalized log-likelihood is given by

$$\begin{aligned} L_{cp}\left(\boldsymbol{\Theta}|\boldsymbol{Y}_1,\ldots,\boldsymbol{Y}_N\right) = \sum_{i=1}^{N}\sum_{g=1}^{G} Z_{gi}\Big[&\log\pi_g + \log\psi\left(\boldsymbol{\gamma}_{ig}, 0, \boldsymbol{\Gamma}\right) \\ &+ \log\psi\left(\boldsymbol{Y}_i; \boldsymbol{S}_i\left(\boldsymbol{\mu}_g + \boldsymbol{\gamma}_{ig}\right), \sigma_e^2\boldsymbol{I}\right)\Big] \\ &-\lambda_L \sum_{1\le g\le g'\le G} \tilde{\boldsymbol{m}}^T|\boldsymbol{\mu}_g - \boldsymbol{\mu}_{g'}| + \lambda_s \sum_{g=1}^{G} \boldsymbol{\mu}_g^T \boldsymbol{W}\boldsymbol{\mu}_g. \end{aligned} \tag{16}$$

At each iteration $t = 0, 1, 2, \ldots$, the EM algorithm consists of the maximization of the expected value of L_{cp}, calculated with respect to the joint distribution of $\boldsymbol{Z}_i$ and $\boldsymbol{\gamma}_{ig}$, given $\boldsymbol{Y}_1,\ldots,\boldsymbol{Y}_N$ and the current parameter estimates $\hat{\boldsymbol{\Theta}}^{(t)} = \{\hat{\pi}_g^{(t)}, \hat{\boldsymbol{\mu}}_g^{(t)}, \hat{\boldsymbol{\Gamma}}^{(t)} = \text{diag}\left(\hat{\sigma}_1^{2(t)},\ldots,\hat{\sigma}_q^{2(t)}\right), \hat{\sigma}_e^{2(t)}\}_{g=1,\ldots,G}$. The algorithm stops when a pre-specified stopping condition is met. At each t, the expected value of L_{cp}, as a function of the probability of membership $\pi_1,\ldots,\pi_G$, is then maximized by setting

$$\hat{\pi}_g^{(t+1)} = \frac{1}{N}\sum_{i=1}^{N} \hat{\pi}_{g|i}^{(t+1)},$$

with $\hat{\pi}_{g|i}^{(t+1)} = \text{E}\left(Z_{ig} = 1|\boldsymbol{Y}_i, \hat{\boldsymbol{\Theta}}^{(t)}\right) = \frac{\hat{\pi}^{(t)}\psi\left(\boldsymbol{Y}_i;\boldsymbol{S}_i\hat{\boldsymbol{\mu}}_g^{(t)},\hat{\boldsymbol{\Sigma}}_i^{(t)}\right)}{\sum_{g'=1}^{G}\hat{\pi}_{g'}^{(t)}\psi\left(\boldsymbol{Y}_i;\boldsymbol{S}_i\hat{\boldsymbol{\mu}}_g^{(t)},\hat{\boldsymbol{\Sigma}}_i^{(t)}\right)}$. Then, L_{cp} is maximized with respect to $\sigma_1^2,\ldots,\sigma_q^2$ by

$$\hat{\sigma}_j^{2(t+1)} = \frac{1}{N}\sum_{i=1}^{N}\sum_{g=1}^{G} \hat{\pi}_{g|i}^{(t+1)}\,\text{E}\left(\gamma_{ig(j)}^2|\boldsymbol{Y}_i, Z_{gi} = 1, \hat{\boldsymbol{\Theta}}^{(t)}\right) \quad j = 1,\ldots,q,$$

where $\gamma_{ig(j)}^2$ indicates the jth entry of $\boldsymbol{\gamma}_{ig}^2$. The value of $\text{E}\left(\gamma_{ig(j)}^2|\boldsymbol{Y}_i, Z_{gi} = 1, \hat{\boldsymbol{\Theta}}^{(t)}\right)$ can be calculated by using the property that the (conditional) distribution of $\boldsymbol{\gamma}_{ig}$,

given $\boldsymbol{Y}_i, Z_{gi} = 1, \hat{\boldsymbol{\Theta}}^{(t)}$, is Gaussian with mean $\hat{\boldsymbol{\Gamma}}^{(t)} \boldsymbol{S}_i^T \left(\boldsymbol{S}_i \hat{\boldsymbol{\Gamma}}^{(t)} \boldsymbol{S}_i^T + \boldsymbol{I} \hat{\sigma}^{2(t)} \right)^{-1} \left(\boldsymbol{Y}_i - \boldsymbol{S}_i \hat{\boldsymbol{\mu}}_g^{(t)} \right)$ and covariance $\hat{\boldsymbol{\Gamma}}^{(t)} - \hat{\boldsymbol{\Gamma}}^{(t)} \boldsymbol{S}_i^T \left(\boldsymbol{S}_i \hat{\boldsymbol{\Gamma}}^{(t)} \boldsymbol{S}_i^T + \boldsymbol{I} \hat{\sigma}^{2(t)} \right)^{-1} \boldsymbol{S}_i \hat{\boldsymbol{\Gamma}}^{(t)}$. Then, σ_e^2 is updated as

$$\hat{\sigma}_e^{2(t+1)} = \frac{1}{\sum_{i=1}^N n_i} \sum_{i=1}^N \sum_{g=1}^G \Bigg[\hat{\pi}_{g|i}^{(t+1)} \left(\boldsymbol{Y}_i - \boldsymbol{S}_i \hat{\boldsymbol{\mu}}_g^{(t)} - \boldsymbol{S}_i \hat{\boldsymbol{\gamma}}_{ig}^{(t)} \right)^T \left(\boldsymbol{Y}_i - \boldsymbol{S}_i \hat{\boldsymbol{\mu}}^{(t)} - \boldsymbol{S}_i \hat{\boldsymbol{\gamma}}_{ig}^{(t)} \right) - \boldsymbol{S}_i \operatorname{Cov} \left(\boldsymbol{\gamma}_{ig} | \boldsymbol{Y}_i, Z_{gi} = 1, \hat{\boldsymbol{\Theta}}^{(t)} \right) \boldsymbol{S}_i^T \Bigg],$$

where $\hat{\boldsymbol{\gamma}}_{ig}^{(t)} = \mathrm{E} \left(\boldsymbol{\gamma}_{ig} | \boldsymbol{Y}_i, Z_{gi} = 1, \hat{\boldsymbol{\Theta}}^{(t)} \right)$.

Finally, the mean vectors $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_G$ that maximize the conditional expectation of L_{cp} are the solution of the following optimization problem

$$\begin{aligned} \hat{\boldsymbol{\mu}}_1^{(t+1)}, \ldots, \hat{\boldsymbol{\mu}}_G^{(t+1)} &= \operatorname{argmin}_{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_G} \frac{1}{2} \sum_{i=1}^N \sum_{g=1}^G \hat{\pi}_{g|i}^{(t+1)} \frac{1}{\hat{\sigma}_e^{(t)}} \left(\boldsymbol{Y}_i - \boldsymbol{S}_i \left(\boldsymbol{\mu}_g + \hat{\boldsymbol{\gamma}}_{ig}^{(t)} \right) \right)^T \\ &\quad \times \left(\boldsymbol{Y}_i - \boldsymbol{S}_i \left(\boldsymbol{\mu}_g + \hat{\boldsymbol{\gamma}}_{ig}^{(t)} \right) \right) \\ &\quad + \lambda_L \sum_{1 \le g \le g' \le G} \tilde{\boldsymbol{m}}^T |\boldsymbol{\mu}_g - \boldsymbol{\mu}_{g'}| + \lambda_s \sum_{g=1}^G \boldsymbol{\mu}^T \boldsymbol{W} \boldsymbol{\mu}_g. \end{aligned} \tag{17}$$

This optimization problem is, unfortunately, a difficult task because of the non differentiability of the absolute value function in zero, and, it has not a closed form solution. However, following the idea of Fan and Li (2001), it can be solved by means of the standard local quadratic approximation method, i.e., by iteratively solving the following quadratic optimization problem for $s = 0, 1, 2, \ldots$

$$\begin{aligned} &\hat{\boldsymbol{\mu}}_1^{(t+1,s+1)}, \ldots, \hat{\boldsymbol{\mu}}_G^{(t+1,s+1)} \\ &= \operatorname{argmin}_{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_G} \frac{1}{2} \sum_{i=1}^N \sum_{g=1}^G \hat{\pi}_{g|i}^{(t+1)} \frac{1}{\hat{\sigma}_e^{(t)}} \left(\boldsymbol{Y}_i - \boldsymbol{S}_i \left(\boldsymbol{\mu}_g + \hat{\boldsymbol{\gamma}}_{ig}^{(t)} \right) \right)^T \\ &\times \left(\boldsymbol{Y}_i - \boldsymbol{S}_i \left(\boldsymbol{\mu}_g + \hat{\boldsymbol{\gamma}}_{ig}^{(t)} \right) \right) \\ &+ \lambda_L \sum_{1 \le g \le g' \le G} |\boldsymbol{\mu}_g - \boldsymbol{\mu}_{g'}|^T \boldsymbol{D}^{(s)} |\boldsymbol{\mu}_g - \boldsymbol{\mu}_{g'}| + \lambda_s \sum_{g=1}^G \boldsymbol{\mu}_g^T \boldsymbol{W} \boldsymbol{\mu}_g, \end{aligned} \tag{18}$$

where $\boldsymbol{D}^{(s)}$ is a diagonal matrix with diagonal entries $\frac{\tau_2 - \tau_1}{2|\hat{\mu}_{g1}^{(t+1,s)} - \hat{\mu}_{g'1}^{(t+1,s)}||\tilde{\mu}_{g1} - \tilde{\mu}_{g'1}|}, \ldots,$ $\frac{\tau_{q+1} - \tau_q}{2|\hat{\mu}_{gq}^{(t+1,s)} - \hat{\mu}_{g'q}^{(t+1,s)}||\tilde{\mu}_{gq} - \tilde{\mu}_{g'q}|}$, and $\hat{\boldsymbol{\mu}}_1^{(t+1,0)} = \hat{\boldsymbol{\mu}}_1^{(t)}, \ldots, \hat{\boldsymbol{\mu}}_G^{(t+1,0)} = \hat{\boldsymbol{\mu}}_G^{(t)}$. Note that, Eq.

(18) is based on the following approximation (Fan and Li 2001)

$$|\mu_{gi} - \mu_{g'i}| \approx \frac{|\mu_{gi} - \mu_{g'i}|^2}{2|\hat{\mu}_{gq}^{(t+1,s)} - \hat{\mu}_{g'q}^{(t+1,s)}|} + \frac{1}{2}|\hat{\mu}_{gq}^{(t+1,s)} - \hat{\mu}_{g'q}^{(t+1,s)}|. \tag{19}$$

The solution to the original problem in Eq. (17) can be satisfactorily approximated by the solution at iteration s^* of the optimization problem in Eq. (18) when a pre-specified stopping condition is met, i.e., $\hat{\boldsymbol{\mu}}_1^{(t+1)} = \hat{\boldsymbol{\mu}}_1^{(t+1,s^*)}, \ldots, \hat{\boldsymbol{\mu}}_G^{(t+1)} = \hat{\boldsymbol{\mu}}_G^{(t+1,s^*)}$. For numerical stability, a reasonable suggestion is to set a lower bound on $|\hat{\mu}_{gi}^{(t+1,s)} - \hat{\mu}_{g'i}^{(t+1,s)}|$, and then to shrink to zero all the estimates below the lower bound. It is worth noting that the proposed modification to the algorithm of James and Sugar (2003) falls within the class of the ECM algorithms (Meng and Rubin 1993). Based on the convergence property of the ECM algorithms, which also holds for the local quadratic approximation in variable selection problems (Fan and Li 2001; Hunter and Li 2005), the proposed algorithm can be proved to converge to a stationary point of the penalized log-likelihood in Eq. (15).

2.4 Data driven parameter selection

The proposed SaS-Funclust method requires the choice of several hyper-parameters viz., the number of clusters G, tuning parameters λ_s, λ_L, dimension q and order k of the set of B-spline functions $\boldsymbol{\Phi}$ as well as knot locations. A standard choice for $\boldsymbol{\Phi}$ is the cubic B-splines (i.e., $k = 4$) with equally spaced knot sequence, which enjoys the optimal property of interpolation (De Boor et al. 1978). As stated in Sect. 2.3, the dimension q should be set as large as possible to reduce, to the greatest possible extent, the approximation error in Eq. (13). This facilitates the estimated cluster means to successfully capture the local feature of the true cluster means. Unfortunately, the larger the value of q, the higher the complexity of the model in Eq. (2), i.e., the number of parameters to estimate. The presence of the smoothness penalty on $\boldsymbol{\mu}_g$, as well as the constraint imposed on $\boldsymbol{\Gamma}_g$, allows one to control the complexity of the model and, thus, to prevent over-fitting issues. The choice of G, λ_s, and λ_L may be based on a K-fold cross-validation procedure. Based on observations divided into K equal-sized disjoint subsets $f_1, \ldots, f_k, \ldots, f_K$, hyper-parameters G, λ_s, and λ_L are chosen as the maximizers of the following function

$$CV\,(G, \lambda_s, \lambda_L) = \frac{1}{K}\sum_{k=1}^{K}\sum_{i \in f_k} \log \sum_{g=1}^{G} \hat{\pi}_g^{-f_k} \psi\left(\boldsymbol{Y}_i; \boldsymbol{S}_i \hat{\boldsymbol{\mu}}_g^{-f_k}, \hat{\boldsymbol{\Sigma}}_i^{-f_k}\right), \tag{20}$$

where $\hat{\pi}_g^{-f_k}$, $\hat{\boldsymbol{\mu}}_g^{-f_k}$ and $\hat{\boldsymbol{\Sigma}}_i^{-f_k}$ denote, respectively, the SaS-Funclust estimates of π_g, $\boldsymbol{\mu}_g$ and $\boldsymbol{\Sigma}_i$ obtained by leaving out the observations of the k-th subset f_k. Usually, the CV function is numerically maximized by means of the classic grid search method (Hastie et al. 2009), that is, an exhaustive searching over a specified grid of hyper-parameter values (Bergstra and Bengio 2012). As in the multivariate regression

setting, the uncertainty of the CV function in estimating the log-likelihood observed for an out-of-sample observation is taken into account by means of the so-called m-standard deviation rule (Hastie et al. 2009). This heuristic rule suggests picking up the most parsimonious model among those achieving values of the CV function that are no more than m standard errors below the maximum of Eq. (20). Note that, in this problem, parsimony is reflected in large λ_s, λ_L and small G. By elaborating on the m-standard deviation rule, we propose to choose G for each value of λ_s, λ_L, with $m = m_1$; secondly, at fixed G, to choose λ_s for each λ_L, with $m = m_2$; thirdly, to choose λ_L at fixed λ_s and G, by using $m = m_3$. In this way, the estimated model is not unnecessarily complex and achieves predictive performance that is comparable to that of the best model (i.e., the one that maximizes the CV function in Eq. (20)). In the Monte Carlo simulation of Sect. 3 and the real-data examples of Sect. 4, the proposed method is implemented with $q = 30$, $K = 5$, $m_1 = m_3 = 0.5$, and $m_2 = 0$. The values of m_1 and m_3 ensure parsimony in the choice of λ_L and G, even though the m-standard deviation is not applied for picking λ_s. In the supplementary information S3, the sensitivity of the SaS-Funclust performance to the choice of q, m_1, m_2, and m_3 has been included. Results show that the m-standard deviation rule is needed to obtain interpretable clustering results and the dimension q may influence the clustering performance as well as the SaS-Funclust ability to detect the informative portions of the domain. As a remark, although the component-wise procedure proposed to choose λ_s, λ_L and G proves itself to be very effective, we recommend whenever possible to directly plot and inspect the CV curve as a function of G, λ_s, and λ_L and to use any information available from the specific application.

The K-fold cross-validation procedure, although may be regarded as a bottleneck of the SaS-Funclust method, is an embarrassingly parallel procedure (Herlihy and Shavit 2011) as the hyper-parameter search can be easily separated into tasks that can be executed concurrently. Embarrassingly parallel procedures are ideals to be performed on a collection of computer servers (Mitrani 2013). Thus, the computational time of the proposed method refers to the time to obtain the clustering results at fixed hyper-parameter values, because the hyper-parameter search can be easily executed in parallel. At the end of Sect. 3 the computational time of the SaS-Funclust method is studied.

3 Simulation study

In this section, the performance of the SaS-Funclust method is compared with competing methods that have already appeared in the literature before, by means of an extensive Monte Carlo simulation study. In particular, we refer to the method proposed by Giacofci et al. (2013) as curvclust, and to that proposed by Bouveyron and Jacques (2011) as funHDDC. These methods are implemented through the homonymous R packages curvclust (Giacofci et al. 2012) and funHDDC (Schmutz and Bouveyron 2019), whereas the SaS-Funclust method is implemented through the R package sasfunclust. In addition, we consider competing methods also the so-called filtering approaches, which are based on two main steps. The first step consists of the estimation of the functions g_i, introduced also in Eq. (1), by means of either smoothing B-splines

or functional principal component analysis (Ramsay and Silverman 2005). The second step aims to apply standard clustering algorithms, viz. hierarchical, k-means and finite mixture model clustering methods (Everitt et al. 2011) to either the resulting B-spline coefficients or the functional principal components scores. Filtering approaches based on the hierarchical, k-means and finite mixture model clustering methods applied to smoothing B-splines coefficients will be hereinafter referred to as B-HC, B-KM and B-FMM, respectively. Whereas, methods based on the hierarchical, k-means and finite mixture model clustering methods applied to the functional principal component decomposition are referred to as FPCA-HC, FPCA-KM and FPCA-FMM, respectively. Finally, we evaluate also the method presented by Ieva et al. (2013), which is referred to as DIS-KM and it basically consists of the application of the k-means clustering to the L^2 distances among the observed curves. Unfortunately, the method of James and Sugar (2003) could not be implemented through the original code (http://faculty.marshall.usc.edu/gareth-james/Research/fclust.txt) due to the high dimensionality of the considered simulated datasets. However, note that the proposed method coincides with the method of James and Sugar (2003) when $\lambda_s = \lambda_L = 0$, which is, thus, implemented in the simulation through the sasfunclust package and referred to as Funclust. Although the SaS-Funclust and Funclust methods are expected to perform similarly, the former should be able to provide much more interpretable clustering partitions. The number of clusters is selected through the Bayesian information criterion (BIC) for the curvclust and funHDDC methods, as suggested by Giacofci et al. (2013) and Bouveyron and Jacques (2011), respectively; whereas the silhouette index (Rousseeuw 1987) is used for the DIS-KM method. The majority rule applied to several validity indices (Charrad et al. 2014) is used to determine the number of clusters for all the filtering approaches. The number of clusters for the Funclust method is obtained through the cross-validation based procedure described in Sect. 2.4. The SaS-Funclust method is implemented as described in Sect. 2 where the initial values of the parameters for the ECM algorithm are chosen by applying the k-means algorithm on the coefficients estimated through smoothing B-splines.

The performance of the clustering procedures in selecting the proper number of clusters and identifying the clustering structure, when the true number of clusters is known, is assessed separately. In particular, the former is measured through the number of selected clusters, whereas the latter is compared through the adjusted Rand index (Hubert and Arabie 1985) denoted by $aRand$. This index accounts for the agreement between true data partitions and clustering results corrected by chance, based on the number of paired objects that are either in the same group or in different groups in both partitions. The $aRand$ yields values between 0 and 1. The larger the value, the higher the similarity between the two corresponding partitions. Moreover, the performance in recovering the true cluster means is measured through the average root mean squared error, calculated as $RMSE = \left[\frac{1}{G}\sum_{g=1}^{G}\int_{\mathcal{T}}\left(\mu_g(t) - \hat{\mu}_g(t)\right)^2 dt\right]^{1/2}$, where $\hat{\mu}_g$ are the estimated cluster means. Whereas, the ability to detect the informative portions of the domain is quantified through the average fractions of correctly identified noninformative portions of the domain.

Three different scenarios are analysed with data generated from $G_t = 2, 3, 4$ clusters and referred to as Scenario I, II and III, respectively. For each scenario, the

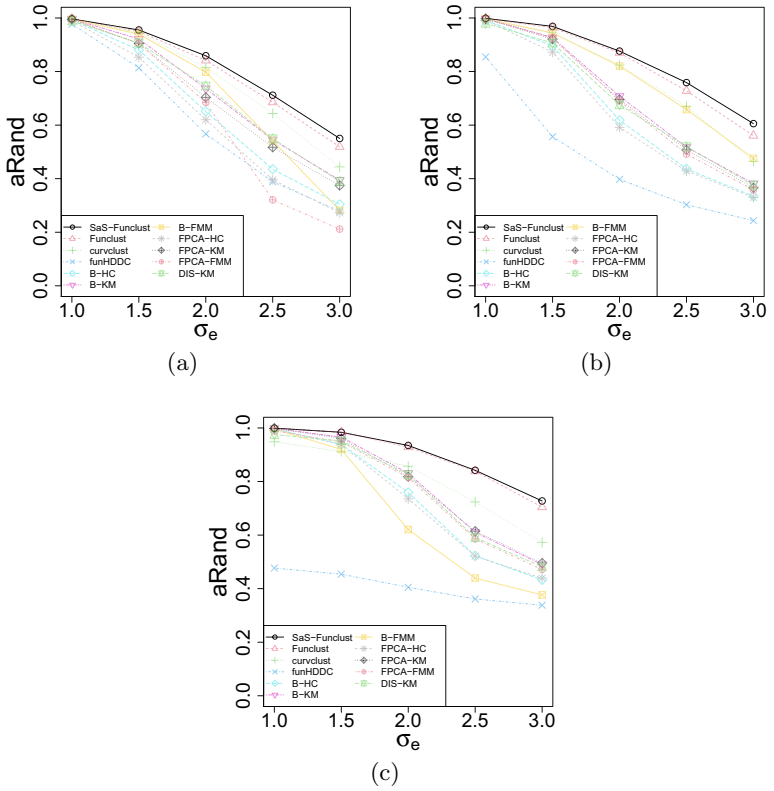

Fig. 2 Average $aRand$ index for **a** Scenario I, **b** Scenario II, and **c** Scenario III as a function of σ_e when the true number of clusters is known

considered methods are evaluated by assessing the performance over 100 independently simulated datasets where measurement errors are generated with five different values of standard error $\sigma_e = 1, 1.5, 2, 2.5, 3$. In all scenarios, the portion of the domain that is noninformative for *all* cluster pairs decreases, whereas the number of portions of the domain that are informative for *specific* cluster pairs increases. Further details about the data generation process and additional simulation results are provided in the supplementary information S2 and S4.

Figure 2 shows the average $aRand$ index values for Scenario I, through III as a function of the standard error σ_e. In Scenario I, at small values of σ_e, all methods perform comparably and provide clustering partitions with $aRand$ very close to 1, which corresponds to the perfect cluster identification. As σ_e increases, the SaS-Funclust method turns out to be the best method and is closely followed, as expected, by the Funclust method, which, differently from the proposed method, does not penalize either the smoothness or the pairwise differences between cluster means. The B-FMM performs also very well, except for $\sigma_e = 3.0$. In Scenario II and III, the SaS-Funclust method is still the best, followed by the Funclust, curvclust and B-FMM in Scenario II and only by Funclust and curvclust methods in Scenario III. Note that in these scenarios,

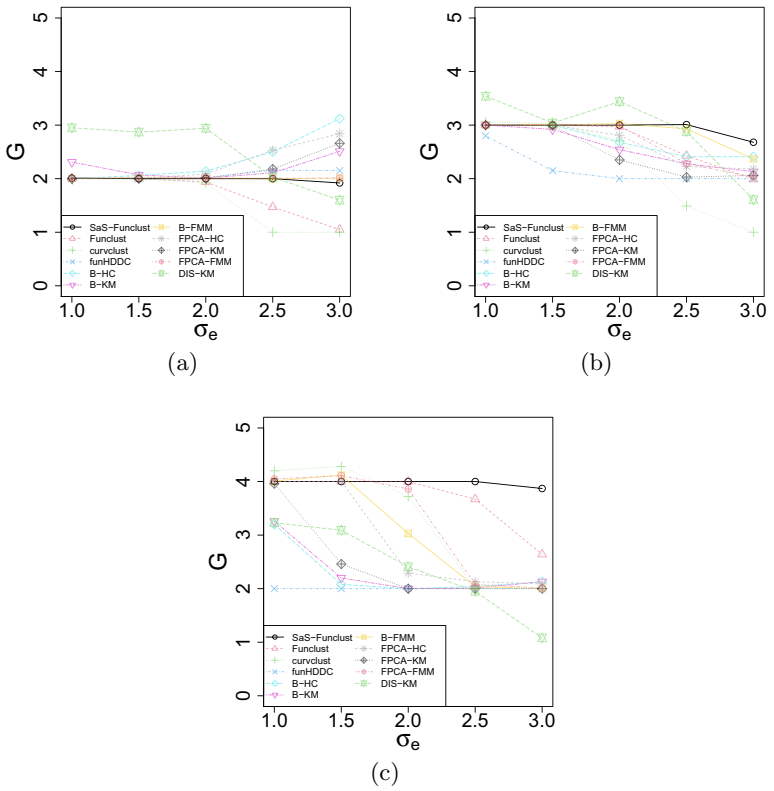

Fig. 3 Average number of selected clusters G for **a** Scenario I, **b** Scenario II, and **c** Scenario III as a function of σ_e

the DIS-KM underperforms also in the most favourable cases as a consequence of the lesser capacity of the L^2 distance to recover the true clustering structure.

Figure 3 shows the average number of selected clusters in all scenarios. It is clear that the SaS-Funclust method is able to identify the true number of clusters much better than the competitors in all considered scenarios. In particular, Scenario II highlights that, especially for large measurement error σ_e, the competing methods reduce their complexity and select, on average, a number of clusters smaller than the true number of clusters $G_t = 3$. This is evident in Scenario III, where the competing methods select, on average, a number of clusters $G = 2$ for $\sigma_e = 2.5, 3.0$, which is smaller than $G_t = 4$.

Figure 4 and Table 1 highlight the ability of the SaS-Funclust method in recovering the true cluster means and detecting the informative portions of the domain. The $RMSE$ is plotted in Fig. 4 for each method as a function of σ_e in all three scenarios. By this figure, the SaS-Funclust method outperforms the competitors in each scenario, especially for large measurement errors, even though the Funclust and curvclust methods show comparable performance.

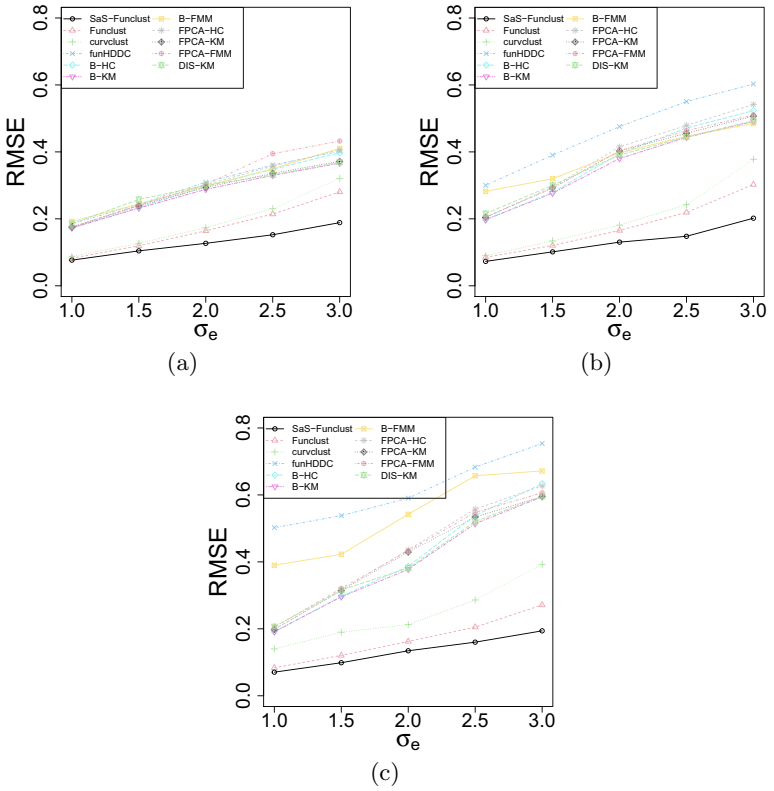

Fig. 4 Average root mean squared error ($RMSE$) for **a** Scenario I, **b** Scenario II, and, **c** Scenario III as a function of σ_e

Table 1 Average fractions of correctly identified noninformative portions of the domain by the SaS-Funclust method for each σ_e and scenario

		Scenario I	Scenario II	Scenario III
σ_e	1.0	0.9956	0.9901	0.9782
	1.5	0.9921	0.9844	0.9627
	2.0	0.9846	0.9589	0.9389
	2.5	0.9565	0.9373	0.8942
	3.0	0.8821	0.8760	0.8024

Table 1 reports the average fractions of correctly identified noninformative portions of the domain for the SaS-Funclust method. This feature is considered only for the proposed method because all the competing non-sparse methods always achieve average fractions of correctly identified noninformative portions of the domain that is equal to zero. In more detail, each entry of the table is obtained as the mean of the average fraction of correctly identified noninformative portions of the domain, over the 100 generated datasets, for each pair of clusters, weighted by the size of the corresponding true noninformative portions of the domain. In Scenario I, it trivially coincides with the average, because the true number of clusters is $G_t = 2$. The proposed method

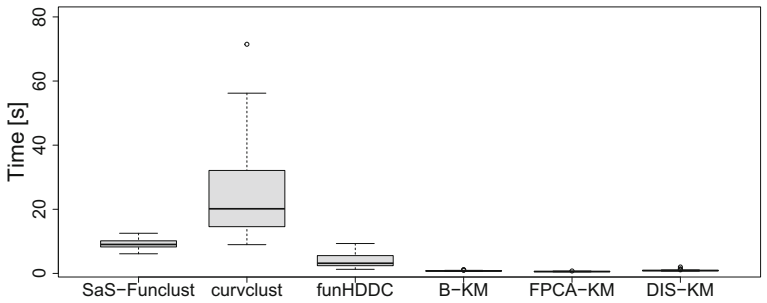

Fig. 5 Computational times for 100 randomly generated datasets from Scenario I of the simulation study with $\sigma_e = 2.0$ and 40 observations for each cluster

is clearly able to provide an interpretable clustering. The fraction of correctly identified noninformative portions of the domain is almost larger than or equal to 0.90 for $\sigma_e \leq 2.5$ and decreases to 0.80 for $\sigma_e = 3.5$. It is worth noting that when $\sigma_e = 1.0$, the pairs of clusters in each scenario are correctly fused over almost all the noninformative portion of the domain in terms of mean differences. This confirms what is shown in Fig. 1 of Sect. 1.

Figure 5 shows computational times needed by a notebook equipped with an Intel®Xenon®CPU E5-1650 v2 @3.50GHz to apply the proposed and competing methods to 100 randomly generated datasets from Scenario I with $\sigma_e = 2.0$ and 40 observations for each cluster. For a fair comparison, computational time does not include operations that can be easily computed in parallel. For instance, the SaS-Funclust computational time is obtained by fixing hyperparameters λ_s and λ_t to their optimal values. The Funclust method implemented as a special case of SaS-Funclust is not reported as it roughly coincides with that of the SaS-Funclust method. The filtering approaches B-HC, B-KM and B-FMM have comparable computational times as well as FPCA-HC, FPCA-KM and FPCA-FMM. Therefore, we report in Fig. 5 results achieved by B-KM and FPCA-KM alone as representative of the two respective groups of filtering approaches, together with the DIS-KM approach.

The proposed method turns out to require larger computational time than the filtering approaches, which are implemented through optimized R packages. However, although the computational convenience, these underperform the proposed method in terms of clustering results. Whereas, the SaS-Funclust algorithm is faster than the curvclust, which already showed worse clustering performance. However, the present implementation of the SaS-Funclust method, even showing adequate computational performance in view of the nice properties imposed on the final solution, has not been highly optimized and leaves room for computational improvement in future research.

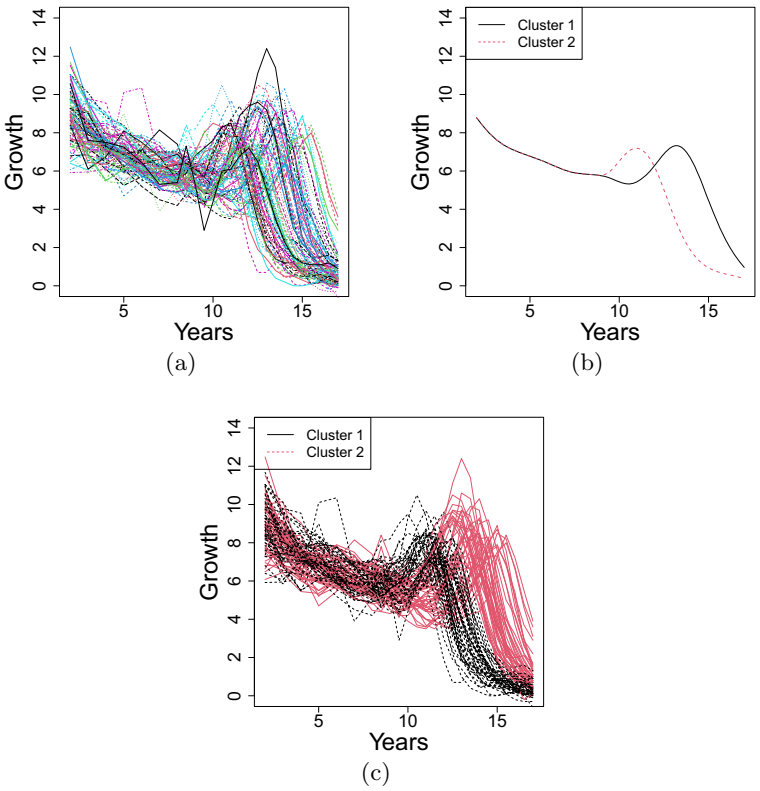

Fig. 6 **a** Growth velocities of 54 girls and 39 boys in the Berkeley growth study dataset; **b** estimated cluster curve means and **c** curve clusters for the SaS-Funclust method in the Berkeley growth study dataset

4 Real-data examples

4.1 Berkeley growth study data

In this section, the SaS-Funclust method is applied to the growth dataset from the Berkeley growth study (Tuddenham 1954), which is available in the R package fda (Ramsay et al. 2020). In this study, the heights of 54 girls and 39 boys were measured 31 times at ages 1 through 18. The aim of the analysis is to cluster the growth curves and compare the results with the partition based on gender differences. This problem has been already addressed by Chiou and Li (2007), Jacques and Preda (2013), Floriello and Vitelli (2017). In particular, we focus on the growth velocities from age 2 to 17, whose discrete values are estimated through the central differences method applied to the growth curves. Figure 6 a shows the interpolating growth velocity curves for all the individuals.

In view of the analysis objective, the clustering methods described in Sect. 3 are applied by setting $G = 2$. As shown in the first row of Table 2, all clustering methods, excluding the B-HC, perform similarly in terms of the $aRand$ index with respect to the

Table 2 The values of the *aRand* index for all the clustering methods with respect to gender difference grouping and the SaS-Funclust partition for the Berkeley growth study dataset

	SaS-Funclust	Funclust	curvclust	funHDDC	B-HC	B-KM	B-FMM	FPCA-HC	FPCA-KM	FPCA-FMM	DIS-KM
Gender difference grouping	0.58	0.58	0.51	0.61	0.20	0.58	0.58	0.58	0.58	0.58	0.58
SaS-Funclust	–	1.00	0.83	0.96	0.37	1.00	1.00	1.00	1.00	1.00	1.00

gender difference partition. Moreover, by looking at the second row of Table 2, which shows the *aRand* index with respect to the SaS-Funclust partition, the competing methods provide partitions very similar to that provided by the SaS-Funclust method.

As expected, the SaS-Funclust method allows for a more interpretable analysis. Figure 6 shows (b) the estimated cluster means and (c) the clustered growth curves for the SaS-Funclust method. The estimated cluster means are fused over the first portion of the domain, whereas they are separated over the remaining portions. This implies that the two identified clusters are not different on average over the first portion of the domain which can be, thus, regarded as noninformative. The separation between the two groups arises over the remaining informative portion of the domain, where two sharp peaks of growth velocity arise, instead. The latter peaks are known in the medical literature as pubertal spurts, in which respect the attained results indicate two main timing/duration groups. In particular, the male pubertal spurt happens later and lasts longer than the female one. Nevertheless, some individuals show unusual growth patterns that are not captured by the cluster analysis. Additionally, the estimated cluster means from the competing methods, not shown here, do not allow for a similar straightforward interpretation.

4.2 Canadian weather data

The Canadian weather dataset contains the daily mean temperature curves, measured in Celsius degrees, recorded at 35 cities in Canada. The temperature profiles are obtained by averaging over the years 1960 through 1994. This is a well-known benchmark dataset available in the R package fda (Ramsay et al. 2020) that has been already studied by Ramsay and Dalzell (1991), Ramsay and Silverman (2005), Sun et al. (2018), Centofanti et al. (2022), Jadhav and Ma (2020). Figure 7a displays the interpolating profiles, where, for computational reasons, temperature curves are sampled every five days.

The ultimate goal of the cluster analysis applied to these curves is the geographical interpretation of the results. In particular, all methods analysed in Sect. 3, are applied by setting $G = 4$ in order to try to recover the grouping of 4 climate zones, viz., Atlantic, Pacific, Continental, Arctic (Jacques and Preda 2013). The first row of Table 3 shows the *aRand* index of the resulting clusters calculated with respect to the 4-climate-zone grouping.

Although the SaS-Funclust, Funclust and the B-HC methods achieve the largest *aRand* in this case, note that *aRand* is inadequately low in all cases, which indicates the clustering structure disagrees with such grouping. The second row of Table 3 reports the *aRand* index for all the competing methods calculated with respect to the SaS-Funclust method. As expected, the proposed clustering agrees with filtering methods based on B-splines and Funclust, while mostly disagreeing with the others.

In terms of interpretability, Fig. 7 shows (b) the estimated cluster means and (c) the geographical distribution of the curves in the clusters obtained by the SaS-Funclust method. From Fig. 7b, the estimated means for clusters 1, 2 and 4 are shown to fuse approximately from day 100 through 250. This is strong evidence that the mean temperature in this period of the year is not significantly different among zones in clusters

Table 3 The values of the *a Rand* index for all the clustering methods with respect to climate zone grouping and the SaS-Funclust partition for the Canadian weather dataset

	SaS-Funclust	Funclust	curvclust	funHDDC	B-HC	B-KM	B-FMM	FPCA-HC	FPCA-KM	FPCA-FMM	DIS-KM
Climate zone grouping	0.37	0.37	0.24	0.21	0.38	0.21	0.33	0.30	0.22	0.17	0.27
SaS-Funclust	–	1.00	0.50	0.35	0.86	0.35	0.93	0.72	0.59	0.43	0.40

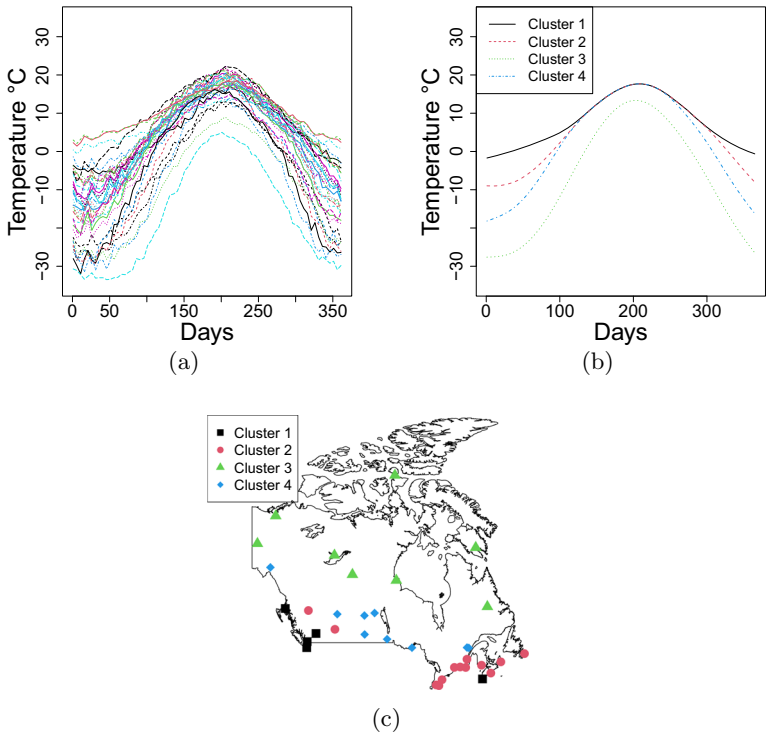

Fig. 7 **a** Daily mean temperature profiles at 35 cities in Canada over the year in the Canadian weather dataset; **b** estimated cluster curve mean and **c** geographical displacement of the curves pertaining to clusters obtained through SaS-Funclust method

1, 2 and 4. Hence, this portion of the domain turns out to be noninformative for the separation of these clusters, whereas the mean temperature is different for the rest of the year. A different pattern is followed by the curves in cluster 3, which shows significantly smaller mean temperatures all over the year. The geographical displacement of the temperature profiles, which are coloured by the clusters identified through the SaS-Funclust method, is reported in Fig. 7c. Observations in clusters 1, 2 and 3 correspond to Pacific, Atlantic and southern continental stations and show similar mean temperature patterns only over the middle days of the year. Observations in cluster 3, which correspond to northern stations, show lower mean temperatures. This nice and plausible interpretation of this well-known real-data example is not possible by means of any competing method.

4.3 ICOSAF project data

The ICOSAF project dataset contains 538 dynamic resistance curves (DRCs), collected during resistance spot welding lab tests at Centro Ricerche Fiat in 2019. The DRCs are collected over a regular grid of 238 points equally spaced by 1 ms. Further details on this dataset can be found in Capezza et al. (2021) and the data are publicly available

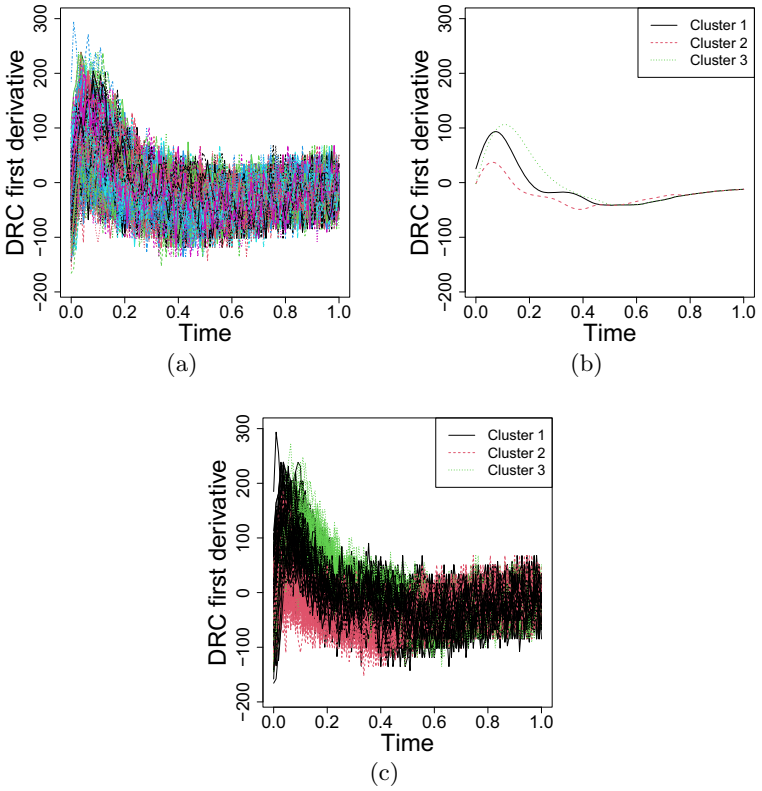

Fig. 8 **a** First derivatives of the 538 DRCs; **b** estimated cluster curve means and **c** curve clusters for the SaS-Funclust method in the ICOSAF project dataset

online at https://github.com/unina-sfere/funclustRSW/. In this example, we focus on the first derivative of the DRCs, estimated by means of the central differences method applied to the DRC values sampled each 2 ms. Figure 8 a shows the first derivative of the DRCs defined, without loss of generality, on the domain [0, 1]. In this setting, the aim of the analysis is to cluster DRCs and identify homogeneous groups of spot welds that share common mechanical and metallurgical properties. Different from the previous datasets, no information is available about a reasonable partition of the DRCs. Therefore, based on the considerations provided by Capezza et al. (2021) as well as on cluster number selection methods that are described for the SaS-Funclust and competing methods in Sects. 2.4 and 3, respectively, we set $G = 3$. Table 4 shows the *aRand* index obtained for all method pairs with respect to the SaS-Funclust partition.

In this case, the SaS-Funclust method provides partitions that are more similar to those obtained through the FPCA-based methods than those obtained with the B-splines filtering approaches. However, the clusters identified by the SaS-Funclust method do not resemble those of the other methods, except for Funclust, as expected.

Table 4 *aRand* index calculated on the ICOSAF project dataset for all competing method partitions with respect to the SaS-Funclust

	Funclust	Curvclust	FunHDDC	B-HC	B-KM	B-FMM	FPCA-HC	FPCA-KM	FPCA-FMM	DIS-KM
SaS-Funclust	0.00	0.00	0.46	0.41	0.35	0.46	0.44	0.27	0.55	0.56

For this dataset, even if results are not reported here, the partition obtained by curvclust differs dramatically from the others and does not provide meaningful clusters.

In this case, also, the SaS-Funclust method allows for an insightful interpretation of the results. The estimated cluster means and the corresponding clustered curves obtained through the SaS-Funclust method, which are displayed in Fig. 8b and c, confirm the ability of the proposed method to fuse cluster means, as it is very clear over the second part of the domain.

In particular, the mean of clusters 1 and 3 are fused from 0.5 to 1, which accounts for the comparable decreasing rate of the DRCs over these clusters. Differently, the mean of cluster 2 is fused with other cluster means between 0.8 and 1, only. This indicates that, between 0.5 and 0.8, DRCs of cluster 3 decrease at a rate that is different from that of DRCs of other clusters. Differences between cluster 2 and clusters 1 and 3 are plainly visible also in the first part of the domain, where DRCs of cluster 2 show lower average velocity. Note also that DRCs of cluster 2 reach their peaks (i.e., zeros of the first derivative) earlier than those of clusters 3 and 1.

5 Conclusions and discussions

This article presented the SaS-Funclust method, a new approach to the sparse clustering of functional data. Differently from methods that have already appeared in the literature before, it was shown to be capable of successfully detecting where cluster pairs are separated. In many applications, this involves limited portions of the domain, which are referred to as informative, and thus, the proposed method allows for more accurate and interpretable cluster analysis. The SaS-Funclust method can be considered as belonging to the model-based clustering procedures with parameters of a general functional Gaussian mixture model estimated by maximizing a penalized version of the log-likelihood function. The key element is the functional adaptive pairwise fusion penalty that, by locally shrinking mean differences, allows pairs of cluster means to be exactly equal over portions of the domain where cluster pairs are not well separated, referred to as noninformative. In addition, a smoothness penalty is introduced to further improve cluster interpretability. The penalized log-likelihood function was maximized by means of a specifically designed expectation-conditional maximization algorithm, and parameter selection was addressed through a cross-validation technique. An extensive Monte Carlo simulation study showed the favourable performance of the proposed method over several competing methods both in terms of clustering accuracy and interpretability. Lastly, the application to real-data examples further demonstrated the practical advantages of the proposed method, which provided, thanks to its sparseness property, new insightful and interpretable solutions to cluster analysis. In the Berkeley growth study example, the SaS-Funclust method highlighted that growth velocity curves of boys and girls show different pubertal spurt, which happens later and lasts longer for males than females. Whereas, in the Canadian weather example, the mean temperatures over the Pacific, Atlantic and southern continental regions were found to be equal over the middle days of the year and different otherwise. Moreover, the proposed method was applied to the ICOSAF project dataset, where, differently from the previous datasets, no information is available about a reasonable

partition. In this case, the SaS-Funclust method identified homogeneous groups of spot welds that showed differences in the rate of change of dynamic resistance curves during the first part of the process only. Such differences are likely to be responsible for distinct mechanical and metallurgical properties of the corresponding spot welds.

As closing remarks, we can envisage several important extensions to refine the proposed method. Regarding the structure of the functional clustering model, the assumption of a common diagonal coefficient covariance matrix across all clusters may be too restrictive in some cases and result in a poor fit. However, more flexible covariance structures dramatically increase the number of parameters to be estimated, already enlarged to achieve sparseness in the SaS-Funclust method. For this reason, the regularization framework shall necessarily be addressed to avoid overfitting, possibly either by constraining the covariance structure, as done in this article, or by means of shrinkage estimators. Unfortunately, the choice of the best approach still remains not straightforward. Furthermore, the covariance structure of the measurement errors could be modified to include more complex relationships, and the model can be extended also by including covariates (James and Sugar 2003).

When the assumption of equal covariance matrices across all clusters is too restrictive, portions of the domain could be informative also in terms of covariance functions, and the SaS-Funclust method may be extended through the integration of proper pairwise penalties applied to the covariance functions. The proposed method is instead based on the assumption that clusters are separated only by their mean values in accordance with the multivariate clustering literature (Xie et al. 2008; Pan and Shen 2007; Wang and Zhu 2008; Guo et al. 2010). Under this assumption, the SaS-Funclust is specifically designed to detect the informative portion of the domain in terms of mean differences. Indeed, the data are assumed to follow a Gaussian mixture distribution (Eq. (5)) with equal and diagonal coefficient covariance matrices across all clusters. However, to the best of the authors' knowledge, the concept of an informative portion of the domain in terms of covariance is completely new both in the functional and the multivariate setting and thus deserves a separate and standalone investigation to overcome methodological and computational difficulties. For instance, the assumption of different coefficient covariance matrices across all clusters would unbearably increase the number of parameters to be estimated as well as the number of hyper-parameters to explore grows exponentially in the K-fold cross-validation procedure described in Sect. 2.4.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s00362-023-01408-1.

Acknowledgements The present work was developed within the activities of the project ARS01_00861 "Integrated Collaborative Systems for Smart Factory (ICOSAF)" coordinated by CRF (Centro Ricerche Fiat Scpa, www.crf.it) and financially supported by MIUR (Ministero dell'Istruzione, dell'Universitá e della Ricerca). We also acknowledge the CINECA award under the ISCRA initiative, for the availability of

high-performance computing resources and support.

Funding None.

Data Availability All data generated or analysed during this study are included in this published article.

Code Availability Software to implement the SaS-Funclust method is available in the R package sasfunclust on CRAN. R code to reproduce graphics and results over competing methods in the simulation study is available as supplementary information.

Declarations

Conflicts of interest The authors declare that they have no conflict of interest.

References

Abraham C, Cornillon PA, Matzner-Lober E et al (2003) Unsupervised curve clustering using b-splines. Scand J Stat 30(3):581–595

Bergstra J, Bengio Y (2012) Random search for hyper-parameter optimization. J Mach Learn Res 13(10):281–305

Bouveyron C, Jacques J (2011) Model-based clustering of time series in group-specific functional subspaces. Adv Data Anal Classif 5(4):281–300

Capezza C, Centofanti F, Lepore A et al (2021) Functional clustering methods for resistance spot welding process data in the automotive industry. Appl Stoch Model Bus Ind 37(5):908–925

Centofanti F, Fontana M, Lepore A et al (2022) Smooth lasso estimator for the function-on-function linear regression model. Comput Stat Data Anal 176(107):556

Charrad M, Ghazzali N, Boiteau V et al (2014) Nbclust an R package for determining the relevant number of clusters in a data set. J Stat Softw 61(6):1–36

Chen H, Reiss PT, Tarpey T (2014) Optimally weighted l2 distance for functional data. Biometrics 70(3):516–525

Chiou JM, Li PL (2007) Functional clustering and identifying substructures of longitudinal data. J R Stat Soc Ser B 69(4):679–699

Cremona MA, Chiaromonte F (2022) Probabilistic k-means with local alignment for clustering and motif discovery in functional data. J Comput Graph Stat. https://doi.org/10.1080/10618600.2022.2156522

De Boor C, De Boor C, Math'ematicien EU et al (1978) A practical guide to splines, vol 27. Springer, New York

Di Iorio J, Vantini S (2019) funbi: a biclustering algorithm for functional data. MOX-Report No 46/2019

Everitt BS, Landau S, Leese M et al (2011) Cluster analysis. Wiley, Hoboken

Fan J, Li R (2001) Variable selection via nonconcave penalized likelihood and its oracle properties. J Am Stat Assoc 96(456):1348–1360

Floriello D, Vitelli V (2017) Sparse clustering of functional data. J Multivar Anal 154:1–18

Friedman JH, Meulman JJ (2004) Clustering objects on subsets of attributes (with discussion). J R Stat Soc 66(4):815–849

Giacofci M, Lambert-Lacroix S, Marot G et al (2012) curvclust: curve clustering. https://CRAN.R-project.org/package=curvclust, R package version 0.0.1

Giacofci M, Lambert-Lacroix S, Marot G et al (2013) Wavelet-based clustering for mixed-effects functional models in high dimension. Biometrics 69(1):31–40
Guo J, Levina E, Michailidis G et al (2010) Pairwise variable selection for high-dimensional model-based clustering. Biometrics 66(3):793–804
Hastie T, Tibshirani R, Friedman J (2009) The elements of statistical learning data mining, inference, and prediction. Springer, New York
Herlihy M, Shavit N (2011) The art of multiprocessor programming. Morgan Kaufmann, Burlington
Horvath L, Kokoszka P (2012) Inference for functional data with applications. Springer, New York
Hsing T, Eubank R (2015) Theoretical foundations of functional data analysis, with an introduction to linear operators. Wiley, Hoboken
Hubert L, Arabie P (1985) Comparing partitions. J Classif 2(1):193–218
Hunter DR, Li R (2005) Variable selection using mm algorithms. Ann Stat 33(4):1617
Ieva F, Paganoni AM, Pigoli D et al (2013) Multivariate functional clustering for the morphological analysis of electrocardiograph curves. J R Stat Soc 62(3):401–418
Jacques J, Preda C (2013) Funclust a curves clustering method using functional random variables density approximation. Neurocomputing 112:164–171
Jacques J, Preda C (2014) Functional data clustering: a survey. Adv Data Anal Classif 8(3):231–255
Jadhav S, Ma S (2020) Functional measurement error in functional regression. Can J Stat 48(2):238–258
James GM, Sugar CA (2003) Clustering for sparsely sampled functional data. J Am Stat Assoc 98(462):397–408
Kokoszka P, Reimherr M (2017) Introduction to functional data analysis. CRC Press, Boca Raton
Maugis C, Celeux G, Martin-Magniette ML (2009) Variable selection for clustering with gaussian mixture models. Biometrics 65(3):701–709
McLachlan GJ, Peel D (2004) Finite mixture models. Wiley, Hoboken
Meng XL, Rubin DB (1993) Maximum likelihood estimation via the ECM algorithm: a general framework. Biometrika 80(2):267–278
Mitrani I (2013) Managing performance and power consumption in a server farm. Ann Oper Res 202(1):121–134
Pan W, Shen X (2007) Penalized model-based clustering with application to variable selection. J Mach Learn Res 8(May):1145–1164
Raftery AE, Dean N (2006) Variable selection for model-based clustering. J Am Stat Assoc 101(473):168–178
Ramsay JO, Dalzell C (1991) Some tools for functional data analysis. J R Stat Soc 53(3):539–572
Ramsay JO, Silverman BW (2005) Functional data analysis. Wiley, Hoboken
Ramsay JO, Hooker G, Graves S (2009) Functional data analysis with R and MATLAB. Springer, New York
Ramsay JO, Graves S, Hooker G (2020) fda: Functional Data Analysis. https://CRAN.R-project.org/package=fda, R package version 5.1.5
Rossi F, Conan-Guez B, El Golli A (2004) Clustering functional data with the som algorithm. In: ESANN, pp 305–312
Rousseeuw PJ (1987) Silhouettes a graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math 20:53–65
Schmutz A, Bouveyron JJC (2019) funHDDC: Univariate and Multivariate Model-Based Clustering in Group-Specific Functional Subspaces. https://CRAN.R-project.org/package=funHDDC, R package version 2.3.0
Schumaker L (2007) Spline functions: basic theory. Cambridge University Press, Cambridge
Serban N, Wasserman L (2005) Cats: clustering after transformation and smoothing. J Am Stat Assoc 100(471):990–999
Sun X, Du P, Wang X et al (2018) Optimal penalized function-on-function regression under a reproducing kernel Hilbert space framework. J Am Stat Assoc 113(524):1601–1611
Tuddenham RD (1954) Physical growth of California boys and girls from birth to eighteen years. Univ Calif Publ Child Dev 1:183–364
Vitelli V (2019) A novel framework for joint sparse clustering and alignment of functional data. arXiv:1912.00687
Wang S, Zhu J (2008) Variable selection for model-based high-dimensional clustering and its application to microarray data. Biometrics 64(2):440–448

Witten DM, Tibshirani R (2010) A framework for feature selection in clustering. J Am Stat Assoc 105(490):713–726
Xie B, Pan W, Shen X (2008) Variable selection in penalized model-based clustering via regularization on grouped parameters. Biometrics 64(3):921–930
Zou H (2006) The adaptive lasso and its oracle properties. J Am Stat Assoc 101(476):1418–1429