STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **10**, March 2022, pp 339–351.
Published online in International Academic Press (www.IAPress.org)

A Basis Approach to Surface Clustering

Adriano Zanin Zambom [1,*], Qing Wang [2], Ronaldo Dias [3]

[1]*Department of Mathematics, California State University Northridge, CA, USA*
[2]*Department of Mathematics, Wellesley College, MA, USA*
[3]*Department of Statistics, State University of Campinas, Sao Paulo, Brazil*

Abstract This paper presents a novel method for clustering surfaces. The proposal involves first using natural splines basis functions in a tensor product to smooth the data and thus reduce the dimension to a finite number of coefficients, and then using these estimated coefficients to cluster the surfaces via k-means or spectral clustering. An extension of the algorithm to clustering higher-dimensional tensors is also discussed. We show that the proposed algorithm exhibits the property of strong consistency, with or without measurement errors, in correctly clustering the data as the sample size increases. Simulation studies suggest that the proposed method outperforms the benchmark k-means and spectral algorithm which use the original data. In addition, an EGG real data example is considered to illustrate the practical application of the proposal.

Keywords natural splines, k-means, spectral clustering, surface clustering

AMS 2010 subject classifications 62H30, 62G08

DOI: 10.19139/soic-2310-5070-1486

1. Introduction

Clustering objects in an infinite dimensional space is a challenging task given the complex nature of the data. Although most data on a continuous domain are observed at a finite set or grid, the computational cost may be too high or the direct application of a clustering procedure to the raw data may fail to capture the intrinsic stochasticity of the observations. Examples of such data structures with infinite dimensions include curves, surfaces, and tensors, which are in reality usually observed with errors. The main goal of this paper is to develop a novel clustering procedure for data sets whose elements are surfaces such as bivariate densities. The idea is to first find an approximation of each surface by estimating the matrix (or tensor) of coefficients of a model in a finite dimensional space, thereby lowering the complexity of the data, and then use these coefficients as the new data for a certain clustering method.

Since the seminal paper of [42], introducing hierarchical clustering, and the work of [18] and [19], discussing k-means clustering, developments and adaptations of these classical algorithms have been seen in a wide range of applications, such as in bioinformatics ([2], [11]), clinical psychiatry ([27]), environmental policy ([21]), market segmentation ([43]), medicine ([39]), text mining ([3]), supply management ([4]), and many other areas. The overall goal of these algorithms is to find partitions of the data based on distance metrics between elements. For instance, in (agglomerative) hierarchical clustering one produces a sequence of $n-1$ partitions of the data, starting with n singleton clusters, and then merging the closest clusters together step by step until a single cluster of n units is formed. The iterative k-means clustering algorithm starts with a set of k initial cluster centers given as input

*Correspondence to: Adriano Zanin Zambom (Email: adriano.zambom@csun.edu). Department of Mathematics, California State University Northridge. 18111 Nordhoff St, Northridge, CA, USA.

ISSN 2310-5070 (online) ISSN 2311-004X (print)

based on a starting-point algorithm or previous knowledge. Each element of the data is then assigned a cluster membership in such a way that the within-cluster sum of squares is minimized.

Clustering methods for curves, i.e. functional data clustering, have been explored by several researchers in the past few years. [22], for example, performs clustering for multivariate functional data with a modification of the k-means clustering procedure. In its motivating example, the goal is to find clusters of similar reconstructed and registered ECGs based on their functional forms and first derivatives. For the multivariate functional data, $\mathbf{X}(t) = (X_1(t), \ldots, X_p(t))$ $(p \in \mathbb{Z}^+)$, where t is in a compact subspace of $\mathbb{R}$ (often representing time), [34] generalize the Mahalanobis distance in Hilbert spaces to create a measure of the distance between functional curves and use it to build a k-means clustering algorithm. Their setting is different from our proposed method in that the time t in [34] is the same for all components of the multivariate functional data, while in this paper we allow bivariate functions, such as $\mathbf{X}(t_1, t_2)$, for example. A non-exhaustive list of recent literature that has studied functional data clustering includes [1], [37], [13], [38], [14], [44], [8], [12], [45], [41], [17], [28], [31], [33], [9], [47], [16].

In this paper we are interested in the generalization of clustering methods, such as the k-means algorithm, to surfaces and tensors. We propose using basis functions in a tensor product as an approximation of the observed data, and then applying the estimated coefficients of the basis functions to cluster the surfaces (or tensors) with k-means or spectral clustering algorithm. Simulations show that our proposed method improves the accuracy of clustering compared to the baseline applied directly to the raw vectorized data.

The remainder of the paper is organized as follows. In Section 2 we describe the estimation procedure of the surfaces and the algorithm for clustering surfaces. Section 3 shows some asymptotic results on the strong consistency of the algorithm in correctly clustering the data as the sample size increases. A generalization of this method to tensor products of higher dimensions is discussed in Section 4. In Section 5 we present simulations that assess the finite sample performance of the proposed method in comparison with the benchmark algorithms.

2. Methodology

Our proposed surface clustering method can be described in a general framework that consists of the following two stages:
Stage 1: Obtain a matrix of estimated coefficients for each surface through a basis smoothing method.
Stage 2: Cluster the surfaces by grouping the estimated coefficient matrices via a clustering method.

In what follows, we detail the proposed methodology by considering natural cubic splines as the smoothing method in Stage 1 to approximate each surface and k-means as the algorithm in Stage 2. However, note that the proposal can be implemented in other forms. In particular, we consider spectral clustering ([32]) as the algorithm in Stage 2 as an alternative proposal in the simulation studies in Section 5.

Let $\mathcal{S}^i := \mathcal{S}^i(x, y), (x, y) \in \mathcal{Q}$ be the underlying data generating process of the ith surface $(1 \leq i \leq n)$, where $\mathcal{Q}$ is a compact subset of $\mathbb{R}^2$. Since data are in general discretely recorded and frequently contaminated with measurement errors, denote

$$z_j^i = \mathcal{S}^i(x_j^i, y_j^i) + \epsilon_j^i, \quad (1 \leq j \leq m_i; 1 \leq i \leq n) \tag{1}$$

as the m_i observed values of the ith surface at coordinates (x_j^i, y_j^i), where ϵ_j^i is the measurement error which is assumed to be i.i.d. with mean 0 and constant finite variance σ^2.

There are different ways of representing functions and surfaces using basis functions, such as wavelets ([26]), spline wavelets ([40]), logsplines ([24]), Fourier series, radial basis, B-splines ([7]), among others. In this paper we focus on natural cubic splines, which impose boundary constraints so that the function is linear beyond the boundary. A natural cubic spline with L knots is associated with L basis functions. It generally yields smoother estimation than polynomial splines or B-splines, and therefore avoids unreasonable extrapolation at the boundary ([20]). The theoretical results we establish next are also valid for the other aforementioned dimension reduction methods. Consider R knots in the x-dimension and L knots in the y-dimension, denoted as ζ_r $(1 \leq r \leq R)$ and

η_ℓ $(1 \leq \ell \leq L)$ respectively. Assume that the surface $\mathcal{S}^i(x, y)$ can be well approximated by a natural cubic spline tensor product, defined as

$$s^i(x, y, \boldsymbol{\Theta}) = \sum_{r=1}^{R} \sum_{l=1}^{L} N_{x,r}(x) N_{y,l}(y) \theta^i_{rl},$$

where θ^i_{rl} are coefficients to be estimated, and $N_{x,r}(\cdot)$ and $N_{y,l}(\cdot)$ are natural cubic spline basis functions that generate the spline spaces $\mathbb{S}_1 = \text{span}\{N_{x,1}, \ldots, N_{x,R}\}$ and $\mathbb{S}_2 = \text{span}\{N_{y,1}, \ldots, N_{y,L}\}$ respectively. $N_{x,r}(\cdot)$ and $N_{y,l}(\cdot)$ are defined as

$$\begin{aligned}
N_{x,1}(x) &= N_{y,1}(y) = 1, \\
N_{x,2}(x) &= x, \\
N_{y,2}(y) &= y, \\
N_{x,r+2}(x) &= d_{x,k}(x) - d_{x,R-1}(x) \ (k = 1, \ldots, R-2), \\
N_{y,\ell+2}(y) &= d_{y,\ell}(y) - d_{y,L-1}(y) \ (\ell = 1, \ldots, L-2),
\end{aligned}$$

where

$$\begin{aligned}
d_{x,r}(x) &= \frac{(x - \zeta_r)^3_+ - (x - \zeta_R)^3_+}{\zeta_R - \zeta_r}, \\
d_{y,\ell}(y) &= \frac{(y - \eta_\ell)^3_+ - (y - \eta_L)^3_+}{\eta_L - \eta_\ell}.
\end{aligned}$$

For ease of notation, we use the same degree (i.e. degree=3) and the same vector of knots for the natural spline functions for all the n surfaces ([5], [6]). In this paper we assume R and L are fixed, however, there are several methods in the literature that describe automatic procedures to obtain these values, see for example [40] and [10]. The linear space of functions formed by the product of these two spaces is denoted by $\mathbb{S}_1 \otimes \mathbb{S}_2$. Because the Sobolev space $\mathcal{H}^2 := \left\{ f : \int f^2 + \int (f')^2 + \int (f''^2) < \infty \right\}$ can be well approximated by $\mathbb{S}_1$ or $\mathbb{S}_2$ in their respective domains ([23], [36], [29], [30]), the space of smooth surfaces in $\mathcal{H}^2 \otimes \mathcal{H}^2$ can be well approximated by $\mathbb{S}_1 \otimes \mathbb{S}_2$.

The $R \times L$ matrix of coefficients $\boldsymbol{\Theta}^i = \{\theta^i_{rl}\}_{1 \leq r \leq R; 1 \leq l \leq L}$ for each surface i $(i = 1, \ldots, n)$, observed with measurement errors, as specified in model (1) can be estimated by minimizing the least squares errors

$$\begin{aligned}
\text{vec}(\widehat{\boldsymbol{\Theta}}^i) &= \arg\min_{\boldsymbol{\Theta}} \sum_{j=1}^{m_i} [z^i_j - s^i(x^i_j, y^i_j, \boldsymbol{\Theta})]^2 \\
&= \arg\min_{\boldsymbol{\Theta}} \sum_{j=1}^{m_i} [z^i_j - \boldsymbol{N}^i_x(x^i_j)^T \boldsymbol{\Theta} \boldsymbol{N}^i_y(y^i_j)]^2 \\
&= \arg\min_{\boldsymbol{\Theta}} \sum_{j=1}^{m_i} [z^i_j - (\boldsymbol{N}^i_y(y^i_j) \otimes \boldsymbol{N}^i_x(x^i_j))^T \text{vec}(\boldsymbol{\Theta})]^2 \\
&= (M^{i^T} M^i)^{-1} M^{i^T} \mathbf{z}^i
\end{aligned}$$

where $\text{vec}(\boldsymbol{\Theta}^i)$ is the vectorization of the matrix $\boldsymbol{\Theta}^i$ arranged by columns, $\mathbf{z}^i = (z^i_1, \ldots, z^i_{m_i})^T$, $\boldsymbol{N}^i_x(x^i_j) = (N^i_1(x^i_j), \ldots, N^i_R(x^i_j))^T$, $\boldsymbol{N}^i_y(y^i_j) = (N^i_1(y^i_j), \ldots, N^i_L(y^i_j))^T$, and M^i is the $m_i \times RL$ matrix with its jth row equal to the $1 \times RL$ vector of $(\boldsymbol{N}^i_y(y^i_j) \otimes \boldsymbol{N}^i_x(x^i_j))^T$.

Surface $\mathcal{S}^i$ is hence summarized by the estimated matrix of parameters $\widehat{\boldsymbol{\Theta}}^i$, which will be used as the input features in clustering. Although the vectorization of the parameter matrix $\boldsymbol{\Theta}^i$ exhibits an elegant expression of the least squares solution, it may lead to loss of the information contained in the matrix structure of the estimated parameters, when employing the clustering procedure. The k-means clustering (or other clustering methods) is a

minimization algorithm based on distances between objects. Thus, we propose to convert vec($\widehat{\mathbf{\Theta}}^i$) back to a matrix form by writing

$$\widehat{\mathbf{\Theta}}^i = \text{devec}\{(M^{i^T} M^i)^{-1} M^{i^T} \mathbf{z}^i\}, \tag{2}$$

where "devec" represents de-vectorization, i.e. arranging vec($\widehat{\mathbf{\Theta}}^i$) into an $R \times L$ matrix whose entries correspond to the parameter matrix $\mathbf{\Theta}^i$. This preserves the spatial structure of the columns and rows of the coefficients that correspond to the hills and valleys of the surface. Such spatial structure of the coefficients can be informative when computing the distance between objects, which are based on matrix distance metrics. Next, one aims to find a partition of the set of surfaces $\mathcal{S} = (\mathcal{S}^1, \ldots, \mathcal{S}^n)$ by grouping the set of estimated parameter matrices $\underline{\widehat{\mathbf{\Theta}}}^n = \{\widehat{\mathbf{\Theta}}^1, \ldots, \widehat{\mathbf{\Theta}}^n\}$ so that surfaces in the same cluster have features as similar as possible, and surfaces in different clusters have dissimilar features. That is, for a given number of clusters K, the algorithm searches for the set of cluster centers $\mathbf{c} = \{c_1, \ldots, c_K\}$ that minimizes

$$\frac{1}{n} \sum_{i=1}^{n} \min_{c \in \mathbf{c}} ||\widehat{\mathbf{\Theta}}^i - c||, \tag{3}$$

where each c represents an $R \times L$ matrix with real elements, and $|| \cdot ||$ is an appropriate matrix norm.

The k-means algorithm finds the partition of the surfaces and their cluster centers $\mathbf{c}$ in the following iterative manner.

Step 1: Initialize the partitions by setting $\mathbf{c}^{(0)}$ as $(\widehat{\mathbf{\Theta}}^{\ell_1}, \ldots, \widehat{\mathbf{\Theta}}^{\ell_K})$ $(\ell_1, \ldots, \ell_K \in \{1, \ldots, n\})$, which can be done, for instance, by choosing

(a) the K matrices with random entries in the range of the entries of $\widehat{\mathbf{\Theta}}^i$ $(i = 1, \ldots, n)$,

(b) the K matrices among $\widehat{\mathbf{\Theta}}^i$ $(i = 1, \ldots, n)$ whose distance (norm) is the largest among themselves: first choose the 2 surfaces that are farthest apart, then sequentially choose other surfaces whose average distance to the previously selected ones is the maximum,

(c) the K matrices $(\widehat{\mathbf{\Theta}}^{\ell_1}, \ldots, \widehat{\mathbf{\Theta}}^{\ell_K})$ among $\widehat{\mathbf{\Theta}}^i$ $(i = 1, \ldots, n)$ so that the sum of the distances (norm) from each $\widehat{\mathbf{\Theta}}^i$ to the closest one in $(\widehat{\mathbf{\Theta}}^{\ell_1}, \ldots, \widehat{\mathbf{\Theta}}^{\ell_K})$ is the minimum.

(d) K randomly chosen matrices among $\widehat{\mathbf{\Theta}}^i$ $(i = 1, \ldots, n)$,

(e) the output of a pre-clustering procedure.

Step 2: Assign each surface, i.e. estimated parameter matrix $\widehat{\mathbf{\Theta}}^i$, to the closest cluster center ℓ according to the minimum distance (norm) $||\widehat{\mathbf{\Theta}}^i - \widehat{\mathbf{\Theta}}^\ell||$ $(\ell \in \{\ell_1, \ldots, \ell_k\})$.

Step 3: Compute the new cluster centers $\mathbf{c}^{(1)} = (c_1^{(1)}, \ldots, c_K^{(1)})$, where $c_\ell^{(1)}$ is the mean of the matrices $\widehat{\mathbf{\Theta}}^i$ for all surfaces i allocated to the ℓ-th cluster $(\ell = 1, \ldots, K)$.

Step 4: Repeat Steps 2 and 3 until there are no more changes in the cluster membership assignments.

3. Asymptotic Results

3.1. Strong Consistency without Measurement Errors

In this section we consider the ideal scenario where the entire surface $\mathcal{S}$ is observable without measurement errors. Lemma 1 below shows that, given an approximation of $\mathcal{S}$ by a natural spline tensor (projection onto $\mathbb{S}_1 \otimes \mathbb{S}_2$), the cluster centers $\mathbf{c}^n$ obtained from minimizing equation (3) converge to a unique (optimal) cluster center set $\mathbf{c}^*$, as the number of surfaces n goes to infinity.

Here we use notations similar to those in [25] and [1]. Let $\Pi(\mathcal{S})$ be the unique matrix $\mathbf{\Theta} \in \mathbb{R}^{R \times L}$ such that

$$\inf_{\mathbf{\Theta} \in \mathbb{R}^{R \times L}} ||\mathcal{S} - s(\cdot, \mathbf{\Theta})|| = ||\mathcal{S} - s(\cdot, \Pi(\mathcal{S}))||.$$

That is, $\Pi(\mathcal{S})$ is the projection of the smooth surface space $\mathcal{H}^2 \otimes \mathcal{H}^2$ onto $\mathbb{S}_1 \otimes \mathbb{S}_2$. Let $\mathcal{N}_{\mathbb{R}^{R \times L}}$ and μ denote the Borel σ-filed of $\mathbb{R}^{R \times L}$ and the image measure of P induced by Π. As Π is continuous, $(\mathbb{R}^{R \times L}, \mathcal{N}_{\mathbb{R}^{R \times L}}, \mu)$ is a

probability space. The surface sequence $(\mathcal{S}^1, \ldots, \mathcal{S}^n)$ induces a sequence $\underline{\mathbf{\Theta}}^n = (\mathbf{\Theta}^1, \ldots, \mathbf{\Theta}^n)$ of i.i.d. random matrices $\mathbf{\Theta}^i = \Pi(\mathcal{S}^i) \in \mathbb{R}^{R\times L}$.

Let

$$\begin{aligned} F &= \{\mathbf{c} \subset \mathbb{R}^{R\times L} | \text{card}(\mathbf{c}) \leq K\}, \\ u(\mathbf{\Theta}, \mathbf{c}) &= \min_{c\in\mathbf{c}} ||\mathbf{\Theta} - c||_F, \end{aligned}$$

and denote the objective function of the k-means algorithm as

$$u_n(\underline{\mathbf{\Theta}}^n, \mathbf{c}) = \frac{1}{n}\sum_{i=1}^{n} u(\mathbf{\Theta}^i, \mathbf{c}),$$

for all $\mathbf{\Theta} \in \mathbb{R}^{R\times L}, c \in \mathbb{R}^{R\times L}$, and $\mathbf{c} \in F$. This differs from the classical clustering methods in that it is composed of norms of matrix differences. Using an appropriate matrix norm, we can establish the following result.

Lemma 1
Let $u(\mathbf{c}) = \int_{\mathbb{R}^{R\times L}} u(\mathbf{\Theta}, \mathbf{c})\mu(d\mathbf{\Theta})$ and assume that $\inf\{u(\mathbf{c})|\mathbf{c} \in F\} < \inf\{u(\mathbf{c})|\mathbf{c} \in F, card(\mathbf{c}) < K\}$. Then, the (unique) minimizer $\mathbf{c}^*$ of $u(\cdot)$ exists and there also exists a unique sequence of measurable functions $\mathbf{c}^n$ from $(\Omega, \mathcal{A}, P)$ into $(F, \mathcal{B}_F)$ such that $\mathbf{c}^n(\omega) \subset M_n$ for all $\omega \in \Omega$ and

$$u_n(\underline{\mathbf{\Theta}}^n, \mathbf{c}^n) = \inf_{\mathbf{c}\subset M_n} u_n(\underline{\mathbf{\Theta}}^n, \mathbf{c}) \text{ a.s.},$$

where $\{M_n\}_n$ is an increasing sequence of convex and compact subsets of $\mathbb{R}^{R\times L}$ such that $\mathbb{R}^{R\times L} = \cup_n M_n$. Furthermore, this sequence $\{\mathbf{c}^n\}$ is strongly consistent to $\mathbf{c}^*$ with respect to the Hausdorff metric.

3.2. Strong Consistency with Measurement Errors

Consider the more realistic model specified in equation (1), where the surfaces are actually recorded with some measurement errors. Without loss of generality, assume the number of observations for each surface is the same, denoted by m. Given a set of observations, $\{(x_1^i, y_1^i, z_1^i), \ldots, (x_m^i, y_m^i, z_m^i)\}$, one can estimate the surface $s^i(\cdot, \mathbf{\Theta})$ by the natural spline estimate $\hat{s}^i = s(\cdot, \widehat{\mathbf{\Theta}})$, where $\widehat{\mathbf{\Theta}}$ is least-square estimated natural spline coefficient matrix given in equation (2). Assume that each surface is observed at different grid points over a compact set $[a, b] \times [c, d]$, for some real constants a, b, c, and d $(a < b, c < d)$. Assume also that for each i, $x_1^i, \ldots, x_m^i$ and $y_1^i, \ldots, y_m^i$ are i.i.d. with probability distributions h and g respectively. The following lemma shows that the estimator $\widehat{\mathbf{\Theta}}$ is strongly consistent for $\mathbf{\Theta} = \Pi(\mathcal{S})$, projection of the smooth surface space $\mathcal{H}^2 \otimes \mathcal{H}^2$ onto the space generated by the natural spline bases.

Lemma 2
Assume the spline bases functions $N_{x,1}, \ldots, N_{x,R}$ and $N_{y,1}, \ldots, N_{y,L}$ are linearly independent on the support of h and g respectively. Assume also that the surfaces $\mathcal{S}^i$ belong to the space $\mathbf{S}$ defined as $\mathcal{H}^2 \otimes \mathcal{H}^2$ restricted to bounded variation on $[a, b] \times [c, d]$. Then, $\widehat{\mathbf{\Theta}}$ converges strongly to $\mathbf{\Theta} = \Pi(\mathcal{S})$ when $m \to \infty$ uniformly over space $\mathbf{S}$. As a result, for almost all $\omega \in \Omega$ and all $\mathcal{S} \in \mathbf{S}$, $||\widehat{\mathbf{\Theta}} - \mathbf{\Theta}|| \to 0$ as m goes to infinity.

Theorem 1
If all the assumptions in Lemma 1 and Lemma 2 are satisfied, the errors are i.i.d. with zero mean and finite constant variance, and the errors are independent of surfaces or the design, for every n, if m is sufficiently large, the set $\arg\min_{\mathbf{c}\subset M_n} u_n(\hat{\underline{\mathbf{\Theta}}}^n, \cdot)$ is nonempty. For all $\omega \in \Omega$, let $\hat{\mathbf{c}}^n(\omega)$ be a minimizer of $u_n(\hat{\underline{\mathbf{\Theta}}}^n(\omega), \cdot)$ with the constraint that $\hat{\mathbf{c}}^n(\omega) \subset M_n$, then $\lim_n \lim_m h(\hat{\mathbf{c}}^n, \mathbf{c}^*) = 0$ a.s.

The proofs of Lemma 1, Lemma 2, and Theorem 1 follow steps similar to those in [1]. We include the proof for Theorem 1 in the appendix.

4. Generalization to Tensor Clustering

The framework designed in Section 2 can be generalized to multi-dimensional clustering. Denote $\mathcal{S}^i(\mathbf{x}), \mathbf{x} = (x_1, \ldots, x_d) \in \mathcal{Q}$, as the data generating tensor mechanism, where $\mathcal{Q}$ is a subset of $\mathbb{R}^d$ and d is the tensor dimension ($d \geq 2$). The observed data $(z_i, \mathbf{x}_i)$ is such that

$$z_j^i = \mathcal{S}^i(\mathbf{x}_j^i) + \epsilon_j^i, \quad (1 \leq j \leq m_i; 1 \leq i \leq n). \tag{4}$$

The approximation of surface $\mathcal{S}^i(\mathbf{x})$ is then based on the smooth natural spline d-dimensional tensor product

$$s^i(\mathbf{x}, \boldsymbol{\Theta}) = \sum_{r_1=1}^{R_1} \ldots \sum_{r_d=1}^{R_d} N_{x_1,r_1}(x_1) \ldots N_{x_d,r_d}(x_d)\theta^i_{r_1,\ldots,r_d},$$

where $\theta^i_{r_1,\ldots,r_d}$ are coefficients to be estimated and $N_{x_1,r_1}(\cdot), \ldots, N_{x_d,r_d}(\cdot)$ are the natural splines basis functions that generate the spline spaces
$\mathbb{S}_1 = \text{span}\{N_{x_1,1}, \ldots, N_{x_1,R_1}\}$, ..., $\mathbb{S}_d = \text{span}\{N_{x_d,1}, \ldots, N_{x_d,R_d}\}$ respectively. The array of coefficients $\boldsymbol{\Theta}$ has dimension $\prod_{i=1}^d r_i$, which is the number of parameters to be estimated.

The multi-dimensional space of smooth surfaces in $\mathcal{H}^2 \otimes \ldots \otimes \mathcal{H}^2$ is then approximated by $\mathbb{S}_1 \otimes \ldots \otimes \mathbb{S}_d$. The least squares solution of this model can be written as

$$\begin{aligned}
\text{vec}(\widehat{\boldsymbol{\Theta}}^i) &= \arg\min_{\boldsymbol{\Theta}} \sum_{j=1}^{m_i} [z_j^i - s^i(\mathbf{x}_j^i, \boldsymbol{\Theta})]^2 \\
&= \arg\min_{\boldsymbol{\Theta}} \sum_{j=1}^{m_i} [z_j^i - (\boldsymbol{N}^i_{x_d}(x^i_{dj}) \otimes \ldots \otimes \boldsymbol{N}^i_{x_1}(x^i_{1j}))^T \text{vec}(\boldsymbol{\Theta})]^2 \\
&= (M^{i^T} M^i)^{-1} M^{i^T} \mathbf{z}^i,
\end{aligned}$$

where $\text{vec}(\boldsymbol{\Theta}^i)$ is the vectorization of matrix $\boldsymbol{\Theta}^i$ arranged by columns, $\mathbf{z}^i = (z_1^i, \ldots, z_{m_i}^i)^T$, $\boldsymbol{N}^i_x(x^i_j) = (N^i_1(x^i_j), \ldots, N^i_R(x^i_j))^T$, $\boldsymbol{N}^i_y(y^i_j) = (N^i_1(y^i_j), \ldots, N^i_L(y^i_j))^T$, and M^i is the $m_i \times RL$ matrix with its jth row equal to the $1 \times RL$ vector of $(\boldsymbol{N}^i_y(y^i_j) \otimes \boldsymbol{N}^i_x(x^i_j))^T$. The proposed procedure can be applied to $\widehat{\boldsymbol{\Theta}}^i$ in a similar fashion as in Section 2, except that one needs to employ an appropriate array norm to evaluate the distances between $\widehat{\boldsymbol{\Theta}}^i$ and $\widehat{\boldsymbol{\Theta}}^j$ $(i \neq j; i, j = 1, ..., n)$. We omit the details for the discussion of tensor clustering in this paper.

5. Simulation Study

In this section we investigate the finite sample performance of the proposed method in clustering surfaces through two simulation scenarios. For comparison purposes, we also evaluate the performance of the k-means clustering and spectral clustering as the benchmark procedures. For both the proposed method using k-means in Stage 2 and the benchmark k-means, we chose the initial guess of the cluster centers in the first step of our proposed algorithm as follows: consider the possible initial guesses given by initialization methods (b), (c), and 50 random initialization as described in method (d) of the k-means algorithm in Section 2. From these 53 possible initial guesses, we choose the one whose K matrices of estimated coefficients, when defined as the center of clusters, have the minimum average distance to the objects in the data assigned to their corresponding clusters. In our numerical studies, we use the Frobenius norm.

The first simulation setting, denoted as Scenario 1, concerns two clusters whose cluster centers are the following (probability density) surfaces, each composed of a mixture of Normal distributions:

$$\begin{aligned}
f_1(x, y) &= 0.3\phi\left((x, y); \begin{pmatrix} 0 \\ -3 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 5 \end{pmatrix}\right) + 0.7\phi\left((x, y); \begin{pmatrix} 0 \\ 3 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\right), \\
f_2(x, y) &= 0.3\phi\left((x, y); \begin{pmatrix} 0 \\ -3 \end{pmatrix}, c\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\right) + 0.7\phi\left((x, y); \begin{pmatrix} 0 \\ 3 \end{pmatrix}, c\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\right),
\end{aligned}$$

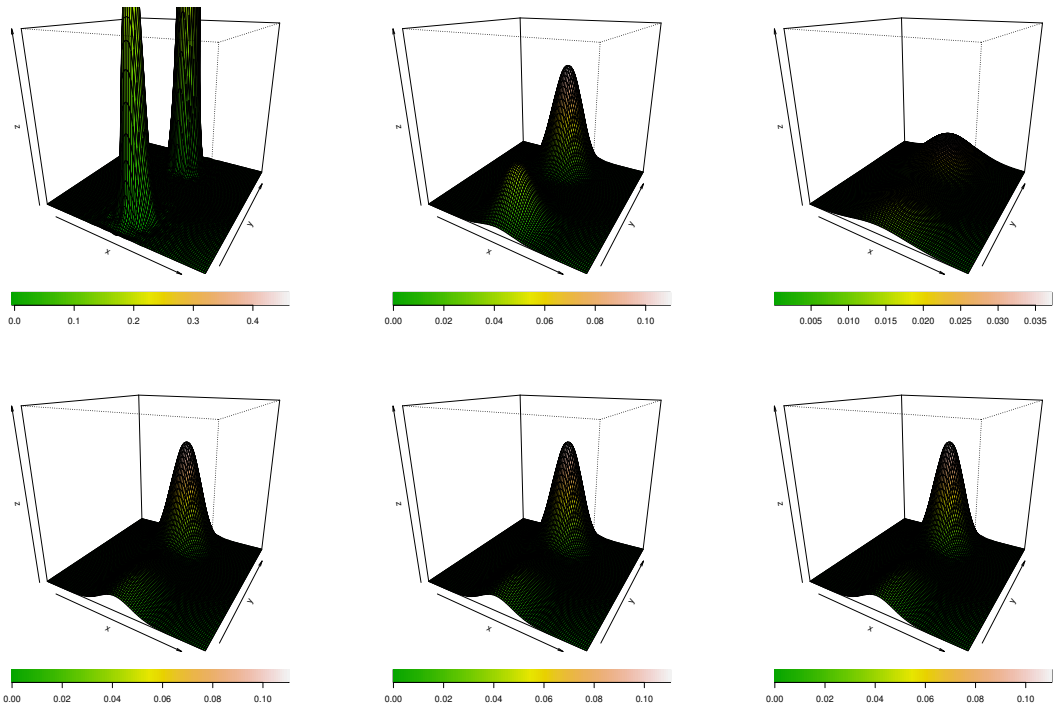

Figure 1. Top: from left to right, surface of the second cluster center (density f_2) with constant c equal to 0.25, 1, and 2.5 respectively. Bottom: surface of the first cluster center (density f_1) for ease of visual comparison with the second cluster center.

where $\phi((x, y); \mu, \Sigma)$ is the probability density function of a bivariate Normal distribution with mean μ and variance Σ, and c is a fixed constant. Note that the covariance matrix of each Normal component of the second cluster center is a multiple of c. For small values of c, the peaks, or modes, of the Normal mixture densities in the second cluster are very high. As c increases, the peaks become lower. The simulations we present below show the performance of the proposed clustering procedure with different values of c, that is, with varying degrees of difficulty in differentiating between the two clusters. Figure 1 shows the centroid surfaces from both clusters without random error (see data generation below), where the top plots display the surface of the second cluster for $c = 0.25, 1$ and 2.5. Note that distinguishing between these two clusters when $c = 1$ is challenging, due to the fact that the only difference is the variance of the first component of the normal mixture. The task of clustering becomes more difficult with the presence of random error in the data generating process which we describe next.

The simulated data (x, y, z) for each surface were generated in a 20×20 grid, i.e. a total of 400 data points, in the square region $(-5, 5) \times (-5, 5)$. The data for Clusters 1 and 2 are generated as follows

$$\text{Cluster 1} : \{(x, y, z) : x_j = y_j = -5 + j/2, j = 1, \ldots, 20; z_j = f_1(x_j, y_j) + \epsilon_j^1\},$$
$$\text{Cluster 2} : \{(x, y, z) : x_j = y_j = -5 + j/2, j = 1, \ldots, 20; z_j = f_2(x_j, y_j) + \epsilon_j^2\},$$

where ϵ_j^1 and ϵ_j^2 are i.i.d. random errors from $N(0, 0.015^2)$ and $N(0, 0.01^2)$ respectively. We generated a total of $n = 60$ surfaces, with 30 samples belonging to each of the two clusters. In the simulations in this paper, we used natural cubic splines basis and a fixed number of 6 knots in each axis for the estimation of the surfaces. This process is repeated for $B = 500$ Monte Carlo simulation runs.

The second simulation setting, denoted as Scenario 2, is composed of 3 clusters whose centroid surfaces are defined by

$$f_3(x,y) = 0.3\phi\left((x,y); \begin{pmatrix}0\\-3\end{pmatrix}, \begin{pmatrix}2&0\\0&1\end{pmatrix}\right) + 0.7\phi\left((x,y); \begin{pmatrix}0\\1\end{pmatrix}, \begin{pmatrix}1&0\\0&1\end{pmatrix}\right),$$
$$f_4(x,y) = 0.3\phi\left((x,y); \begin{pmatrix}0\\1\end{pmatrix}, \begin{pmatrix}2&0\\0&2\end{pmatrix}\right) + 0.7\phi\left((x,y); \begin{pmatrix}0\\-3\end{pmatrix}, \begin{pmatrix}1&0\\0&1\end{pmatrix}\right),$$
$$f_5(x,y) = 0.3\phi\left((x,y); \begin{pmatrix}0\\-2\end{pmatrix}, c\begin{pmatrix}2&0\\0&1\end{pmatrix}\right) + 0.7\phi\left((x,y); \begin{pmatrix}1\\0\end{pmatrix}, c\begin{pmatrix}1&0\\0&1\end{pmatrix}\right).$$

Figure 2 shows the centroid surfaces from the 3 clusters for $c = 0.25, 1$ and 2.5. The first two clusters are easier to distinguish, since they have opposite hills. However, the third cluster may bring a challenge to the task, especially since the data are generated with random errors.

The data for each cluster are simulated as follows:

$$\text{Cluster 1} : \{(x,y,z) : x_j = y_j = -5 + j/2, j = 1,\dots,20; z_j = f_3(x_j,y_j) + \epsilon_j^1\},$$
$$\text{Cluster 2} : \{(x,y,z) : x_j = y_j = -5 + j/2, j = 1,\dots,20; z_j = f_4(x_j,y_j) + \epsilon_j^2\},$$
$$\text{Cluster 3} : \{(x,y,z) : x_j = y_j = -5 + j/2, j = 1,\dots,20; z_j = f_5(x_j,y_j) + \epsilon_j^3\},$$

where $\epsilon_j^1, \epsilon_j^2, \epsilon_j^3$ are i.i.d. random errors from $N(0, 0.015^2)$. We generated a total of $n = 60$ surfaces, with 20 surfaces belonging to each cluster. This process was repeated for $B = 500$ Monte Carlos simulation runs.

It is well known that the initialization of the k-means procedure can have an immense influence on the clustering results ([35], [15]). For this reason and for a fair comparison between the proposal and the benchmark, we initialized each procedure in the same way as described in the first paragraph of this section. The benchmark k-means and spectral clustering apply the algorithm to the vectorized raw data set, while the proposal employs the k-means or spectral clustering to the natural cubic splines estimated coefficients.

In order to evaluate the results we consider the following performance measure. Let $\mathcal{S}^{i(b)}$ denote the randomly generated surface i $(i = 1,\dots,n)$ in the b-th simulation run $(b = 1,\dots,B)$. Let $L(\mathcal{S}^{i(b)}), L^*(\mathcal{S}^{i(b)}) \in \{1,\dots,K\}$ be the predicted and true cluster membership for surface i respectively. Define

$$\phi = \frac{\sum_{b=1}^{B} \min_{\tau\in T} \sum_{i=1}^{n} I(L(\mathcal{S}^{i(b)}) \neq \tau(L^*(\mathcal{S}^{i(b)})))}{B}, \tag{5}$$

where the T in "$\tau \in T$" is the set of permutations over $\{1,\dots,K\}$. It measures the number of mis-specification errors of a clustering procedure, which is based on the MCE measure in [15]. Hence, the performance measure ϕ is the mean mis-specification, i.e., the average number of surfaces assigned to incorrect clusters.

Table 1 shows the results, out of 500 Monte Carlo simulation runs, of the proposed algorithm that is realized by either the k-means or spectral clustering, as well as the results of the benchmark k-means and benchmark spectral clustering. For Scenario 1, for small values of c, the variance of the densities in Cluster 2 is small and hence the hill is high (see top left plot of Figure 1). In this case all algorithms cluster the data satisfactorily. When $c = 1$ or 1.25, the benchmark k-means and benchmark spectral clustering incorrectly specify an average of 11 and 28 curves respectively. In comparison, the proposed methods still keep its error rate. In fact, for all values of c, the proposed procedures are able to maintain a high clustering performance, achieving virtually no errors. For values of c between 1 and 1.5, the two bivariate densities that compose each cluster are very similar (see top and bottom middle plots in Figure 1). In such cases, the benchmark k-means has poor performance and incorrectly clusters a large number of surfaces. For larger values of c, the variance of the densities in Cluster 1 is large, while the hills in Cluster 2 are low, and therefore the difference between the two clusters are again more visible (see top and bottom right plots in Figure 1). For these values of c, all methods under comparison yield satisfactory results. For Scenario 2, in all cases the benchmark k-means algorithm yields the largest mean number of incorrectly clustered surfaces. For values of c between 1.5 and 2, the benchmark spectral clustering performs the best, with the fewest incorrectly

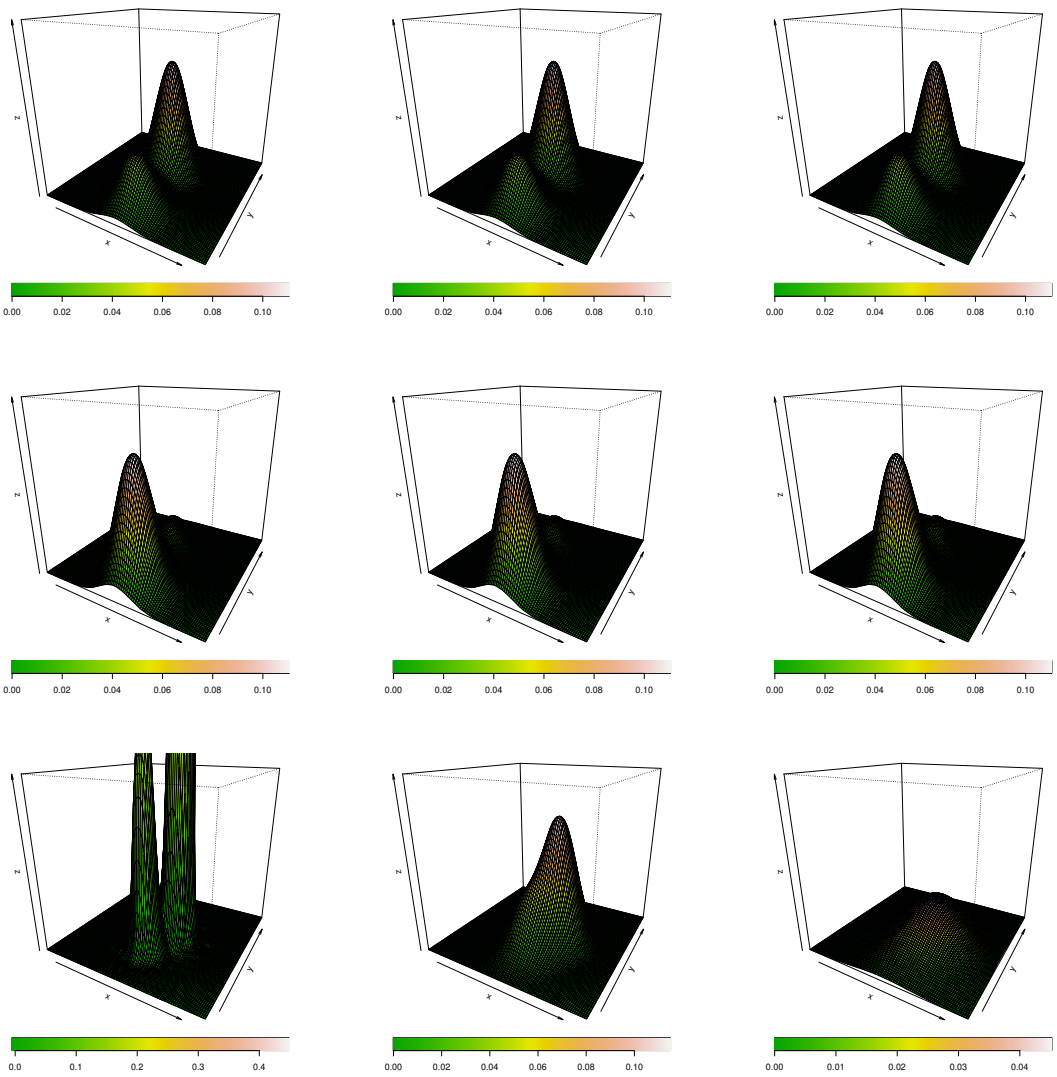

Figure 2. From left to right, surfaces of the cluster center with constant c equal to 0.25, 1, and 2.5 respectively; rows one, two, and three correspond to f_3, f_4, and f_5 respectively.

clustered surfaces. However, for values of c that are smaller than 1.5 or greater than 2, the proposed algorithm, realized by either the k-means or spectral clustering method in Stage 2, outperforms both benchmarks, with fewer incorrectly clustered surfaces.

6. Real Data Analysis: EEG Clustering

In this section we illustrate the proposed surface clustering method in an analysis of the Electroencephalogram (EEG) dataset, which is available at the University of California Irvine Machine Learning Repository (https://archive.ics.uci.edu/ml/datasets/eeg+database). The dataset is composed of 122 subjects that are divided into two groups: alcohol and control. We focus on the case where participants were exposed to two stimuli (S1 and

Table 1. Comparison of mean number of incorrectly clustered surfaces out of 500 Monte Carlo runs, with different values of c in the first simulation setting.

	Scenario 1				Scenario 2			
	Proposal		Benchmark		Proposal		Benchmark	
c	k-means	spectral	k-means	spectral	k-means	spectral	k-means	spectral
.25	0	0	0	0	16.93	15.05	37.84	20
.5	0	0	0	0	17.00	16.00	38.00	20
.75	0	0	0	1	17.00	20.00	38.00	20
1	0	0	11	28	18.00	19.00	38.00	23
1.25	2	0	11	25	20.00	29.00	38.00	22
1.5	0	1	29	7	21.00	15.00	38.00	6
1.75	0	0	27	4	22.00	14.00	38.00	10
2	0	0	27	0	22.00	13.00	38.00	11
2.25	0	0	27	0	21.00	14.00	38.00	25
2.5	0	0	0	0	24.00	16.00	38.00	25

S2). The stimulus was a picture of an object chosen from the 1980 Snodgrass and Vanderwart picture set. They were either matched, where S1 was identical to S2, or not matched, where S1 was different from S2. On the scalp of each subject, 64 electrodes were positioned according to the Standard Electrode Position Nomenclature, American Electroencephalographic Association, and measurements were taken at 256 Hz (3.9-msec epoch) for 1 second (See [46] for details). The channels (i.e. electrodes) and time compose the surface domain (x, y) and the measurements are the response z.

We averaged over all trials for each individual and used the mean surface to represent the output of each subject's response to the stimuli. Visualization of the raw data of a subject in the control group as well as the raw data of a subject in the alcohol group are shown in Figure 3. The EEG surface of the control subject seems to be somewhat

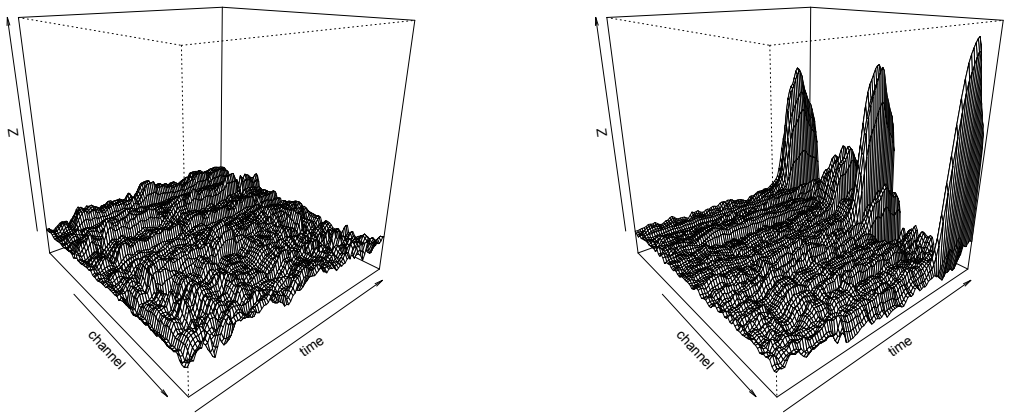

Figure 3. Example of EEG measurements for a subject in the control group (left) and alcohol group (right).

flat with some random noise, while the EEG surface of the subject in the alcohol group shows a few hills for some channels at later time stamps. However, this is not the case for all subjects. It can also be observed that the EEG surfaces of some subjects in the control group have hills at later time stamps, while some subjects in the alcohol group have somewhat flat EEG surfaces throughout time. This imposes practical challenges on the clustering task, as we discuss below.

We applied our proposed clustering method to the full dataset of 122 patients, where the elements clustered by the k-means algorithm were the mean surfaces of the subjects across their trials. Given that we know the actual

groups of the subjects, i.e. either control or alcohol, we can compute how many errors in clustering were made by the proposed algorithm when applied to this dataset. The number of incorrectly clustered subjects was 41 (33.6%) for the matched stimuli, and 38 (31.1%) for the not matched stimuli using the proposal based on k-means and 44 (36.1%) and 39 (32.0%) respectively when using the proposal based on spectral clustering. For comparison purposes, the benchmark k-means obtained 41 (33.6%) and 44 (36.1%) for the matched and not matched stimuli respectively, and the benchmark spectrum obtained 44 (36.1%) and 39 (32.0%) for the matched and not matched stimuli respectively.

7. Conclusions

This article proposes a new approach for clustering surfaces using the coefficients obtained from natural cubic splines that approximate their data generating smooth forms as the basis for a k-means clustering algorithm. The proposed method is shown to be strongly consistent in both stochastic and non-stochastic cases. Compared to the classical k-means procedure applied directly to the vectorized version of the data that may lose the geometric structure of the surface, our surface clustering procedure performs consistently better in correctly clustering surfaces observed with noise in simulation studies. From the wide range of applications of surface clustering, in this paper we studied the identification of effects of alcohol in the brain by clustering Electroencephalogram data from 122 patients and clustering into alcohol and control groups, where the surfaces were defined by the stimuli response at 64 electrodes throughout time stamps.

Acknowledgements

We would like to thank Fapesp for partially funding this research (Fapesp 2018/04654-9, 2017/15306-9, 2019/04535-2).

Appendix

Proof for Theorem 1

In the proof of this theorem, we follow steps similar to those in Theorem 1 of [1]. From Lemma 2, for almost all $\omega \in \Omega$ and all $\mathcal{S} \in \mathbf{S}$, the sequence $u_n(\hat{\underline{\boldsymbol{\Theta}}}^n, \cdot) \to u_n(\underline{\boldsymbol{\Theta}}^n, \cdot)$ as $m \to +\infty$. Therefore, to prove Theorem 1, we need to show the convergence of a sequence of minimizers, $\hat{\mathbf{c}}^n$, of $u_n(\hat{\underline{\boldsymbol{\Theta}}}, \cdot)$ to $\mathbf{c}^n$ which is the minimizer of $u_n(\underline{\boldsymbol{\Theta}}, \cdot)$.

Let K be the number of clusters. Define

$$\begin{aligned} \psi : (\mathbb{R}^{R\times L})^K &\to F = \{\mathbf{c} \subset \mathbb{R}^{R\times L} | \text{card}(\mathbf{c}) \leq K\}, \\ \mathbf{c} &\to \{c_1, \ldots, c_K\} \end{aligned}$$

and

$$\begin{aligned} \Phi_n : (\mathbb{R}^{R\times L})^n \times (\mathbb{R}^{R\times L})^K &\to (-\infty, +\infty], \\ (\underline{\boldsymbol{\Theta}}^n, \mathbf{c}) &\to u_n(\underline{\boldsymbol{\Theta}}^n, \psi(\mathbf{c})) + \mathbb{I}_{K_n}(\psi(\mathbf{c})) \end{aligned}$$

where $K_n = \{\mathbf{c} \in F | \mathbf{c} \subset M_n\}$, and $\mathbb{I}_{K_n}(\mathbf{c}) = 0$ if $\mathbf{c} \in K_n$ and $\mathbb{I}_{K_n} = +\infty$ if $\mathbf{c} \notin K_n$. It can be seen that finding $\arg\min_{\mathbf{c}\subset M_n} u_n(\boldsymbol{\Theta}^n, \mathbf{c})$ is equivalent to finding $\arg\min_{\mathbf{c}\in(\mathbb{R}^{R\times L})^K} \Phi_n(\boldsymbol{\Theta}^n, \mathbf{c})$.

For all $\omega \in \Omega$, define

$$\begin{aligned} g_m(\omega, \cdot) : (\mathbb{R}^{R\times L})^K &\to (-\infty, +\infty] \\ \mathbf{c} &\to \Phi_n(\hat{\underline{\boldsymbol{\Theta}}}^n(\omega), \mathbf{c}) \\ g(\omega, \cdot) : (\mathbb{R}^{R\times L})^K &\to (-\infty, +\infty] \\ \mathbf{c} &\to \Phi_n(\underline{\boldsymbol{\Theta}}^n, \mathbf{c}) \end{aligned}$$

Note that $\hat{\underline{\mathbf{\Theta}}}^n$ depends on the number of observational points m for each surface in the data set. For simplicity of notation, we omit including a subscript that emphasizes the dependency of $\hat{\underline{\mathbf{\Theta}}}^n$ on m.

For the rest of the proof, we want to apply the theorem of convergence in minimization (Kochafeller & Wets, 1998, p. 266). It is stated as follows: if the sequence $(g_m)_m$ is eventually level-bounded and epi-converges to g, with g_m and g being lower semi-continuous and proper, then for m sufficiently large, the sets $\arg\min g_m$ are non-empty and are all included in the same compact set. Furthermore, if $t_m \in \arg\min g_m$ and t is a cluster point of $(t_m)_m$, then $t \in \arg\min g$. To employ the theorem of convergence in minimization, we need to show that (1) $(g_m)_m$ is level-bounded; (2) $g(\omega,\cdot)$ is the epi-limit of $g_m(\omega,\cdot)$; and (3) $g_m(\omega,\cdot)$ and $g(\omega,\cdot)$ are both lower semi-continuous. The proofs for (1)-(3) are given below.

Take $\omega \in \Omega$ such that $\hat{\underline{\mathbf{\Theta}}}^n(\omega) \to \underline{\mathbf{\Theta}}^n(\omega)$ as $n \to +\infty$ (by Lemma 2). Then, for all $\mathcal{S} \in \mathbf{S}$ and for all $\mathbf{c} \in (\mathbb{R}^{R\times L})^K$ and all sequences $(\mathbf{c}_m)_m$ such that $\mathbf{c}_m \to \mathbf{c}$, by the continuity of u_n and ψ, we have

$$\lim_{m\to+\infty} g_m(\omega,\mathbf{c}_m) = g(\omega,\mathbf{c}).$$

Thus, $g(\omega,\cdot)$ is the epi-limit of $g_m(\omega,\cdot)$. In addition, because $g(\omega,\cdot)$ and $g_m(\omega,\cdot)$ take finite values on K_n, they are proper. Furthermore, since $\psi^{-1}(K_n) = \prod_{i=1}^k M_n$ and $\left\{\mathbf{c} \in (\mathbb{R}^{R\times L})^K | g_m(\omega,\mathbf{c}) \le \alpha\right\}$ for all $\alpha \in \mathbb{R}$, it follows that $g_m(\omega,\cdot)$ is level bounded.

Next, we show that both $g_m(\omega,\cdot)$ and $g(\omega,\cdot)$ are lower semi-continuous. First, by the continuity of u_n and ψ, the set

$$\left\{\mathbf{c} \in (\mathbb{R}^{R\times L})^K | g_m(\omega,\mathbf{c}) \le \alpha\right\} = \left\{\mathbf{c} \in (\mathbb{R}^{R\times L})^K | u_n(\hat{\underline{\mathbf{\Theta}}}^n, \psi(\mathbf{c}) \le \alpha\right\} \cap \psi^{-1}(K_n)$$

is closed, and therefore $g_m(\omega,\cdot)$ is lower semi-continuous. Similarly, it can be shown that $g(\omega,\cdot)$ is also lower semi-continuous.

By the theorem of convergence in minimization, we conclude that for large enough m, $\arg\min g_m(\omega,\cdot)$ is non-empty, and every cluster point of a sequence $\mathbf{c}_m(\omega)$ of the minimizer of $g_m(\omega,\cdot)$ is a minimizer of $g(\omega,\cdot)$. Note that

$$\arg\min_{\mathbf{c}\subset M_n} u_n(\hat{\underline{\mathbf{\Theta}}}^n, \mathbf{c}) = \psi(\arg\min g_m(\omega,\cdot)).$$

Hence, for m that is sufficiently large, $\arg\min_{\mathbf{c}\subset M_n} u_n(\hat{\underline{\mathbf{\Theta}}}^n, \mathbf{c})$ is non-empty. If one takes $(\hat{\mathbf{c}}_m^n(\omega))_m$, a sequence of minimizers of $u_n(\hat{\underline{\mathbf{\Theta}}}^n,\cdot)$ in M_n, then there exists a sequence $(\mathbf{a}_m(\omega))_m$ s.t. $\hat{\mathbf{c}}_m^n(\omega) = \psi(\mathbf{a}_m(\omega))$ and $\mathbf{a}_m(\omega)$ is a minimizer of $g_m(\omega,\cdot)$. Let $\mathbf{c}^n(\omega)$ be the unique minimizer of $u_n(\mathbf{\Theta}^n,\cdot)$. We have $\arg\min g(\omega,\cdot) = \psi^{-1}(\mathbf{c}^n(\omega))$ is a finite set, and $(\mathbf{a}_m(\omega))_m$ has a finite number of cluster points. Then, for all $\epsilon > 0$, there exists M s.t. for all $m > M$, $\mathbf{a}_m(\omega)$ is close to a cluster point s.t. $h(\hat{\mathbf{c}}_m^n(\omega), \mathbf{c}^n(\omega)) < \epsilon$ (here $h(\cdot)$ represents the Hausdorff metric). That is, $\hat{\mathbf{c}}_m^n(\omega) \to \mathbf{c}^n(\omega)$ as $n \to +\infty$.

By Lemma 1, we conclude that $\lim_{m\to+\infty}\lim_{n\to+\infty} h(\hat{\mathbf{c}}^n(\omega), \mathbf{c}^*) = 0$ for almost all ω. This completes the proof.

REFERENCES

1. C. Abraham, P. A. Cornillon, E. Matzner-Laber, and N. Molinari. *Unsupervised curve clustering using b-splines*, Scandinavian Journal of Statistics, vol. 30, pp. 581-595, 2003.
2. B. Abu-Jamous, R. Fa, and A. Nandi. *Interactive cluster analysis in bioinformatics*, Wiley, 2015.
3. L. M. Abualigah, A. T. Khader, and E. S. Hanandeh. *A combination of objective functions and hybrid krill herd algorithm for text document clustering analysis*, Engineering Applications of Artificial Intelligence, vol. 73, pp. 111-125, 2018.
4. J. Blackhurst, M. J. Rungtusanatham, K. Scheibe, and S. Ambulkar. *Supply chain vulnerability assessment: A network based visualization and clustering analysis approach*, Journal of Purchasing and Supply Management, vol. 24, pp. 21-30, 2018.
5. C. de Boor. Subroutine package for calculating with b-splines. *Techn.Rep. LA-4728-MS, Los Alamos Sci.Lab, Los Alamos NM*, pp.. 109-121, 1971.
6. C. de Boor. *On calculating with b-splines*, Journal of Approximation Theory, vol. 6, number 1, pp. 50-62, 1972.
7. C. de Boor. *Package for calculating with b-splines*, SIAM Journal on Numerical Analysis, vol. 14, number 3, pp. 441–472, 1977.
8. M. Boullé. *Functional data clustering via piecewise constant nonparametric density estimation*, Pattern Recognition, vol. 45, number 12, pp. 4389-4401, 2012.
9. K. Deng and X. Zhang. *Tensor envelope mixture model for simultaneous clustering and multiway dimension reduction*, Biometrics - online: doi = https://doi.org/10.1111/biom.13486, 2021.

10. R. Dias and D. Gamerman. *A Bayesian approach to hybrid splines nonparametric regression*, Journal of Stat. Comp. and Simul., vol. 72, number 4, pp. 285-297, 2002.
11. A. H. Duran, T. M. Greco, B. Vollmer, C. I. M., K. Crunewald, and M. Topf. *Protein interactions and consensus clustering analysis uncover insights into herpesvirus virion structure and function relationships*, PLOS Biology, 2019.
12. M. Febrero-Bande and M. de la Fuente. *Statistical computing in functional data analysis: The r package fda.usc*, Journal of Statistical Software, vol. 51, number 4, 2012.
13. F. Ferraty and P. Vieu. *Nonparametric functional data analysis*. Springer Series in Statistics, 2006.
14. D. Floriello. *Functional sparse k-means clustering*, Thesis, Politecnico di Milano, 2011.
15. P. Franti and S. Sieranoja. *How much can k-means be improved by using better initialization and repeats?*, Pattern Recognition, vol. 93, pp. 95-112, 2019.
16. C. Frévent, M.-S. Ahmed, M. Marbac, and M. Genin. *Detecting spatial clusters in functional data: New scan statistic approaches*, Spatial Statistics, vol. 46, pp. 100550, 2021.
17. M. L. L. García, R. García-Rodenas, and A. G. Gómez. *K-means algorithms for functional data*, NEUROCOMPUTING, vol. 151, pp. 231-245, 2015.
18. J. A. Hartigan. *Clustering algorithms*. Wiley, 1975.
19. J. A. Hartigan and M. A. Wong. *Algorithm as 136: A k-means clustering algorithm*, JRSSC, vol. 28, number 1, pp. 100-108, 1979.
20. T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*. Springer, 2 edition, 2016.
21. G. Hu, M. Kaur, K. Hewage, and R. Sadiq. *Fuzzy clustering analysis of hydraulic fracturing additives for environmental and human health risk mitigation*, Clearn Technologies and Environmental Policy, vol. 21, pp. 39–53, 2019.
22. F. Ieva, A. M. Paganoni, D. Pigoli, and V. Vitelli. *Multivariate functional clustering for the morphological analysis of electrocardiograph curves*, JRSSC, vol. 62, number 3, pp. 401-418, 2013.
23. S. Karlin. *Some variational problems on certain sobolev spaces and perfect splines*, Bull. Amer. Math. Soc., vol. 79, number 1, pp. 124-128, 01 1973.
24. C. Kooperberg and C. J. Stone. *Logspline density estimation for censored data*, Journal of Computational and Graphical Statistics, vol. 1, number 4, pp. 301-328, 1992.
25. J. Lemaire. *Proprietes asymptotiques en classification*, Statistiques et analyse des donnees, vol. 8, pp. 41-58, 1983.
26. S. Mallat. *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. Academic Press, Inc., USA, 3rd edition, 2008.
27. S. Jiménez-Murcia. *Phenotypes in gambling disorder using sociodemographic and clinical clustering analysis: an unidentified new subtype?*, Front Psychiatry, vol. 10, number 173, 2019.
28. J. Kim and H.-S. Oh. *Pseudo-quantile functional data clustering*, Journal of Multivariate Analysis, vol. 178, pp. 104626, 2020.
29. P. Lachout, E. Liebscher, and S. Vogel. *Strong convergence of estimators as ϵn-minimisers of optimisation problemsof optimisation problems*, Annals of the Institute of Statistical Mathematics, vol. 57, number 2, pp. 291-313, 2005.
30. S. R. Lindemann and S. M. LaValle. *Simple and efficient algorithms for computing smooth, collision-free feedback laws over given cell decompositions*, The International Journal of Robotics Research, vol. 28, number 5, pp. 600-621, 2009.
31. H. Lu, S. Liu, H. Wei, and J. Tu. *Multi-kernel fuzzy clustering based on auto-encoder for fmri functional network*, Expert Systems with Applications, vol. 159, pp. 113513, 2020.
32. U. von Luxburg. *A tutorial on spectral clustering*, Statistics and Computing, vol. 17, number 4, pp. 395-416, 2004.
33. Q. Mai, X. Zhang, Y. Pan, and K. Deng. *A doubly enhanced em algorithm for model-based tensor clustering*, Journal of the American Statistical Association, vol. 0, number 0, pp. 1-15, 2021.
34. A. Martino, A. Ghiglietti, F. Ieva, and A. M. Paganoni. *A k-means procedure based on a mahalanobis type distance for clustering multivariate functional data*, Statistical Methods & Applications, vol. 28, number 2, pp. 301-322, 2019.
35. J.M Pena, J.A Lozano, and P Larranaga. *An empirical comparison of four initialization methods for the k-means algorithm*, Pattern Recognition Letters, vol. 20, number 10, pp. 1027-1040, 1999.
36. U. Reif. *Uniform b-spline approximation in sobolev spaces*, Numerical Algorithms, vol. 15, number 1, pp. 1-14, 1997.
37. T. Tarpey and K. K. J. Kinateder. *Clustering functional data*, Journal of Classification, vol. 20, number 1, pp. 93-114, May 2003.
38. S. Tokushige, H. Yadohisa, and K. Inada. *Crisp and fuzzy k-means clustering algorithms for multivariate functional data*, Computational Statistics, vol. 22, number 1, pp. 1-16, 2007.
39. M. S. Udler, J. Kim, M. von Grotthuss, S. Bons-Guarch, J. B. Cole, J. Chiou, C. D. Anderson, M. Boehnke, M. Laakso, G. Atzmon, J. M. Glaser, B. Mercader, K. Gaulton, J. Flannick, G. Getz, and J. C. Florez. *Type 2 diabetes genetic loci informed by multi-trait associations point to disease mechanisms and subtypes: A soft clustering analysis*, PLOS Medicine, 2018.
40. M. A. Unser. *Ten good reasons for using spline wavelets*, In Wavelet Applications in Signal and Image Processing V, vol. 3169, pp. 422-431, 1997.
41. G. Wang, N. Lin, and B. Zhang. *Functional k-means inverse regression*, Computational Statistics & Data Analysis, vol. 70, number C, pp. 172-182, 2014.
42. J. H. Ward Jr. *Hierarchical grouping to optimize an objective function*, Journal of the American Statistical Association, vol. 58, number 301, pp. 236-244, 1963.
43. M. Wedel and W. Kamakura. *Market segmentation: conceptual and methodological foundations*. Springer Science & Business Media, 2 edition, 1999.
44. M. Yamamoto. *Clustering of functional data in a low-dimensional subspace*, Advances in Data Analysis and Classification, vol. 6, number 3, pp. 219-247, Oct 2012.
45. M. Yamamoto and Y. Terada. *Functional factorial k-means analysis*, Computational Statistics and Data Analysis, vol. 79, pp. 133-148, 2014.
46. X.L. Zhang, H. Begleiter, B. Porjesz, W. Wang, and A. Litke. *Event related potentials during object recognition tasks*, Brain Research Bulletin, vol. 38, number 6, pp. 531-538, 1995.
47. Y. Zhang, X. Bi, N. Tang, and A. Qu. *Dynamic tensor recommender systems*, Journal of Machine Learning Research, vol. 22, pp. 1-35, 2021.