Advances in Data Analysis and Classification (2022) 16:181–199
https://doi.org/10.1007/s11634-021-00482-3

REGULAR ARTICLE

Robust clustering of functional directional data

Pedro C. Álvarez-Esteban[1] · Luis A. García-Escudero[1]

Received: 14 December 2020 / Revised: 10 September 2021 / Accepted: 23 October 2021 /
Published online: 9 December 2021

Abstract

A robust approach for clustering functional directional data is proposed. The proposal adapts "impartial trimming" techniques to this particular framework. Impartial trimming uses the dataset itself to tell us which appears to be the most outlying curves. A feasible algorithm is proposed for its practical implementation justified by some theoretical properties. A "warping" approach is also introduced which allows including controlled time warping in that robust clustering procedure to detect typical "templates". The proposed methodology is illustrated in a real data analysis problem where it is applied to cluster aircraft trajectories.

Keywords Cluster analysis · Robustness · Functional data analysis · Directional data · Warping

Mathematics Subject Classification 62H30 · 62H11 · 62G35

1 Introduction

Modern technologies are increasingly allowing us to measure phenomena continuously in time. In those cases, although the curves are often discretized, data sets can be seen as made of curves rather than finite-dimensional measurements. Functional Data Analysis (Ramsay and Silverman 2005; Ferraty and Vieu 2006) are the set of statistical tools specially developed to deal with this particular type of data. In particular, functional

This work is part of a joint project (AIRPORTS) with Boeing Research and Technology Europe (BRT&E), and is funded by CDTI and MINECO (FEDER) Grant IDI-20150616, CIEN 2015. Authors are partly funded by MINECO (Grants MTM2017-86061-C2-1-P and MTM2017-86061-C2-2-P), and by Consejería de Educación de la JCyL and FEDER funds (Grants VA005P17 and VA002G18).

✉ Pedro C. Álvarez-Esteban
pedrocesar.alvarez@uva.es

Luis A. García-Escudero
lagarcia@eio.uva.es

[1] Dpto. de Estadística e Investigación Operativa, IMUVA, Universidad de Valladolid, Paseo de Belén, 7, 47011 Valladolid, Spain

Cluster Analysis is recently receiving considerable attention, as can be seen in recent review papers as Jacques and Preda (2014), Hitchcock and Greenwood (2015), and Yassouridis and Leisch (2017).

In this work, we will focus on providing a functional clustering approach that can be applied to cluster functional directional data and where robustness also plays an important role. See Mardia and Jupp (2009) and Ley and Verdebout (2017) for some general references on directional statistics.

It is known that (even) a small fraction of contaminating data can be very detrimental in Cluster Analysis (García-Escudero and Gordaliza 1999). This justifies the interest of applying robust clustering techniques that are able to resist certain amount of outlying observations (García-Escudero et al. 2010; Ritter 2015; García-Escudero et al. 2015). Moreover, the application of robust clustering techniques may be also useful in order to detect anomalous features in our data, which can be very interesting once we are able to explain their anomalous behaviour. In this work, robustness is achieved by allowing to discard a proportion α of functional directional data throughout an "impartial" trimming procedure. The term impartial means that it is the data itself the one that tell us which are the most anomalous curves. This impartial trimming approach was introduced in Rousseeuw (1984), Gordaliza (1991) and Cuesta-Albertos et al. (1997).

Impartial trimming has been already applied in Functional Cluster Analysis in García-Escudero and Gordaliza (2005), Cuesta-Albertos and Fraiman (2007) and, more recently, in Rivera-García et al. (2019). The approach adopted now in our work is more closely related with Cuesta-Albertos and Fraiman (2007) because we are not using projections into finite-dimensional functional subspaces.

The proposed methodology will be introduced in Sect. 2 together with some theoretical results for the proper characterization of the optimal solutions to the underlying problems. These theoretical results are latter applied in Sect. 3 to derive a feasible algorithm for the practical application of the methodology.

The proposed algorithm will be extended in Sect. 4 to allow for "time warping" in the curves assigned to each cluster. Time warping is an appealing idea to address misalignment problems within clusters and to detect typical "templates" which are also useful to describe the detected clusters.

Some guidelines about how to make sensible choices for the number of clusters k and for the trimming level α are given in Sect. 5.

Section 6 provides a simple simulation study to illustrate the ability of the proposed methodology to properly recover alignments in unaligned data, and simultaneously trim outlying curves.

Finally, Sect. 7 presents a real data application aimed at clustering aircraft trajectories that motivated our interest in clustering functional directional data. This real data example serves to illustrate all the material introduced in previous sections. Other applications for this methodology are surely possible. For instance, another direct application could be to cluster weather stations based on the observed evolution of local wind directions. Detecting anomalous weather stations, and explaining why they exhibit such strange behavior, can be an interesting task.

2 Methodology

We are going to use the extrinsic distance in the unit sphere $\mathbb{S}^1$ such that, for every pair ω_1 and ω_2 in $[0, 2\pi]$ (or, analogously, ω_1 and ω_2 in $\mathbb{S}^1$), we take

$$d(\omega_1, \omega_2) = 1 - \cos(\omega_1 - \omega_2) \tag{1}$$

to measure their distance. Given $\omega_1, \dots, \omega_n$ in $\mathbb{S}^1$, the directional mean $\overline{\omega}$ is defined as

$$\overline{\omega} = \arg\min_{\omega \in \mathbb{S}^1} \sum_{i=1}^{n} d(\omega_i, \omega).$$

In this work, we are interested in clustering $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_n$ "directional functions" where each $\boldsymbol{\theta}_i$ belongs to $C([0, 1], \mathbb{S}^1)$ as the set of continuous functions defined on $[0, 1]$ and taking values in $\mathbb{S}^1$ when $\mathbb{S}^1$ is equipped with extrinsic distance in (1). This means that we start from a sample $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_n$ in where every $\boldsymbol{\theta}_i$ satisfies

$$\begin{aligned} \boldsymbol{\theta}_i \ :&[0, 1] \to \mathbb{S}^1 \\ t &\mapsto \boldsymbol{\theta}_i(t), \end{aligned}$$

and $\boldsymbol{\theta}_i$ is assumed to be a continuous function in $\mathbb{S}^1$. To simplify notation, we simply use the notation $\mathcal{F}$ for denoting $C([0, 1], \mathbb{S}^1)$.

The extrinsic distance in $\mathbb{S}^1$ can be extended to a distance in the set of directional functions $\mathcal{F}$ just by considering an integrated extrinsic distance defined as:

$$D(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = \int_0^1 d(\boldsymbol{\theta}_1(t), \boldsymbol{\theta}_2(t))dt = \int_0^1 \big(1 - \cos(\boldsymbol{\theta}_1(t) - \boldsymbol{\theta}_2(t))\big)dt. \tag{2}$$

For robust clustering purposes, we consider the impartial trimming approach, where we try that the sample itself inform us which are the "most outlying" directional functions to be trimmed. In this approach, we search for k directional functions $\{\boldsymbol{m}_1, \dots, \boldsymbol{m}_k\} \subset \mathcal{F}$ (with $k << n$) or "prototypes" that better serve to summarize our set of observed curves $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_n$, but allowing a proportion $\alpha \in [0, 1)$ of curves to be trimmed in an "optimal" way. These $\boldsymbol{m}_1, \dots, \boldsymbol{m}_k$ serve to create a partition of the non-trimmed $\boldsymbol{\theta}_i$ directional functions, by assigning each $\boldsymbol{\theta}_i$ to cluster J, if $\boldsymbol{\theta}_i$ is closest to $\boldsymbol{m}_J$ than to the other $\boldsymbol{m}_j$ prototypes when using the distance in (2). The fraction α of trimmed $\boldsymbol{\theta}_i$ directional functions, as suspicious of being outliers, are left unassigned.

To be more precise, let us introduce some further notation in this functional directional framework. For $\boldsymbol{\theta} \in \mathcal{F}$ and

$$M = \{\boldsymbol{m}_1, \dots, \boldsymbol{m}_k\} \subset \mathcal{F},$$

let us define

$$D(\boldsymbol{\theta}; M) = D(\boldsymbol{\theta}; \{\boldsymbol{m}_1, \ldots, \boldsymbol{m}_k\}) := \inf_{j=1,\ldots,k} D(\boldsymbol{\theta}, \boldsymbol{m}_j).$$

The trimmed k-mean problem can be now defined through the following double minimization procedure:

$$\inf_{\mathcal{H}\subset\{1,\ldots,n\}:\mathrm{card}(\mathcal{H})=\lceil n(1-\alpha)\rceil} \inf_{M\subset\mathcal{F}:\mathrm{card}(M)=k} \sum_{i\in\mathcal{H}} D(\boldsymbol{\theta}_i; M), \tag{3}$$

where $\lceil x \rceil$ is the least integer greater than or equal to x.

Note that this double minimization is done on every possible subset of indexes $\mathcal{H}$ such that $\mathcal{H} \subset \{1, \ldots, n\}$ and $\mathrm{card}(\mathcal{H}) = \lceil n(1-\alpha)\rceil$, and every possible set of k function $M = \{\boldsymbol{m}_1, \ldots, \boldsymbol{m}_k\}$ in $\mathcal{F}$. The result of that double minimization is the set of optimally non-trimmed functions, i.e. those with indexes in $\mathcal{H}$, and a set with the k optimal center or prototype directional functions given by $M = \{\boldsymbol{m}_1, \ldots, \boldsymbol{m}_k\}$.

To gain insights on how this complex double maximization can be simplified, let us present some additional notation and two main results. Given k centres $M = \{\boldsymbol{m}_1, \ldots, \boldsymbol{m}_k\}$, let us define the optimal radius as

$$r_\alpha(M) = \inf\{r \geq 0 : \mathrm{card}\{i : D(\boldsymbol{\theta}_i; M) \geq r\} \leq [n\alpha]\}.$$

In other words, if

$$D(\boldsymbol{\theta}_{(1)}; M) \leq D(\boldsymbol{\theta}_{(2)}; M) \leq \ldots \leq D(\boldsymbol{\theta}_{(n)}; M),$$

then $r_\alpha(M) = D(\boldsymbol{\theta}_{(\lceil n(1-\alpha)\rceil)}; M)$.

By using this notation, we introduce the subset $\mathcal{H}(M) \subset \{1, \ldots, n\}$, with $\mathrm{card}(\mathcal{H}(M)) = \lceil n(1-\alpha)\rceil$, defined as

$$\mathcal{H}(M) = \{i : D(\boldsymbol{\theta}_i; M) \leq r_\alpha(M)\}. \tag{4}$$

Theorem 1 show that, if the optimal set M were known, then the optimal set $\mathcal{H}$ including all the non-trimmed curves can be chosen as those with indexes in $\mathcal{H}(M)$. Notice that $\mathcal{H}(M)$ can easily determined from M just by sorting all the $D(\boldsymbol{\theta}_i; M)$ distances.

Theorem 1 *Let $V_{\mathcal{H}}(M) = \sum_{i\in\mathcal{H}} D(\boldsymbol{\theta}_i; M)$. We have that $V_{\mathcal{H}(M)}(M) \leq V_{\mathcal{H}}(M)$ for every $\mathcal{H}$ with card$(\mathcal{H}) = \lceil n(1-\alpha)\rceil$ for $V_{\mathcal{H}(M)}(M)$ as defined in (4).*

Proof Since $\mathrm{card}(\mathcal{H}) = \mathrm{card}(\mathcal{H}(M)) = \lceil n(1-\alpha)\rceil$, we trivially have that

$$\mathrm{card}\{\mathcal{H} \cap \mathcal{H}(M)^c\} = \mathrm{card}\{\mathcal{H}^c \cap \mathcal{H}(M)\}.$$

We have $V_{\mathcal{H}(M)} \leq V_{\mathcal{H}}$ because

$$V_{\mathcal{H}} = \sum_{i\in\mathcal{H}\cap\mathcal{H}(M)} D(\boldsymbol{\theta}_i; M) + \sum_{i\in\mathcal{H}\cap\mathcal{H}(M)^c} D(\boldsymbol{\theta}_i; M)$$

$$
\begin{aligned}
&\geq \sum_{i\in\mathcal{H}\cap\mathcal{H}(M)} D(\boldsymbol{\theta}_i; M) + r_\alpha(M)\cdot \text{card}\{\mathcal{H}\cap\mathcal{H}(M)^c\}\\
&= \sum_{i\in\mathcal{H}\cap\mathcal{H}(M)} D(\boldsymbol{\theta}_i; M) + r_\alpha(M)\cdot \text{card}\{\mathcal{H}^c\cap\mathcal{H}(M)\}\\
&\geq \sum_{i\in\mathcal{H}\cap\mathcal{H}(M)} D(\boldsymbol{\theta}_i; M) + \sum_{i\in\mathcal{H}^c\cap\mathcal{H}(M)} D(\boldsymbol{\theta}_i; M)\\
&= V_{\mathcal{H}(M)}.
\end{aligned}
$$

□

Therefore, if M were known, there is no needed to explore the combinatorial set of all possible $\mathcal{H}$ subsets. Moreover, it is also trivial to see that $\mathcal{H}(M)$ can be optimally split as

$$\mathcal{H}(M) = \mathcal{H}_1(M)\cup ...\cup \mathcal{H}_k(M),$$

with

$$\mathcal{H}_j(M) = \{i \in \mathcal{H}(M) \text{ such that } D(\boldsymbol{\theta}_i; M) = D(\boldsymbol{\theta}_i, \boldsymbol{m}_j)\}$$

(again, only depending on sorting these $D(\boldsymbol{\theta}_i; M)$ distances).

Consequently, if we introduce

$$V(M) = \sum_{j=1}^{k}\sum_{i\in\mathcal{H}_j(M)} D(\boldsymbol{\theta}_i, \boldsymbol{m}_j),$$

then the double minimization in (3) can be rewritten as a single minimization, only depending on the set of k optimal directional functions M, as

$$\inf_{M\subset\mathcal{F}:\text{card}(M)=k} V(M).$$

On the other hand, if we assume $\mathcal{H} = \mathcal{H}_1\cup ...\cup\mathcal{H}_k$ were known, the optimal $\boldsymbol{m}_j$ can be easily obtained by computing pointwise directional means:

Theorem 2 *Let $\mathcal{H} = \mathcal{H}_1\cup ...\cup\mathcal{H}_k$ fixed and let us define*

$$V_{\mathcal{H}}(M) = \sum_{j=1}^{k}\sum_{i\in\mathcal{H}_j} D(\boldsymbol{\theta}_i, \boldsymbol{m}_j)$$

for $M = \{\boldsymbol{m}_1,\ldots,\boldsymbol{m}_k\}$. The minimal value of $V_{\mathcal{H}}(M)$ is attained when

$$t\mapsto \boldsymbol{m}_j(t) = \arg\min_{\omega}\sum_{i\in\mathcal{H}_j} d(\boldsymbol{\theta}_i(t),\omega) = \arg\min_{\omega}\sum_{i\in\mathcal{H}_j}(1-\cos(\boldsymbol{\theta}_i(t)-\omega)),$$

for $t \in [0, 1]$ *and* $j = 1, ..., k$.

Proof This result easily follows from the fact that

$$\sum_{i \in \mathcal{H}_j} D(\boldsymbol{\theta}_i, \boldsymbol{m}_j) = \int_0^1 \left(\sum_{i \in \mathcal{H}_j} d(\boldsymbol{\theta}_i(t), \boldsymbol{m}_j) \right) dt.$$

Given that $d(\cdot, \cdot)$ is positive, the minimization of that integral is done throughout a pointwise minimization of the integrating term. □

Notice that closed expressions for $\boldsymbol{m}_j(t)$ are available as

$$\boldsymbol{m}_j(t) = \arg \left(\frac{\sum_{i \in \mathcal{H}_j} \exp(\mathrm{i} \cdot \boldsymbol{\theta}_i(t))}{\mathrm{card}(\mathcal{H}_j)} \right),$$

(i denotes the imaginary unit) or, analogously,

$$\boldsymbol{m}_j(t) = \mathrm{atan}_2 \left(\frac{1}{\mathrm{card}(\mathcal{H}_j)} \sum_{i \in \mathcal{H}_j} \sin \boldsymbol{\theta}_i(t), \frac{1}{\mathrm{card}(\mathcal{H}_j)} \sum_{i \in \mathcal{H}_j} \cos \boldsymbol{\theta}_i(t) \right).$$

Theorem 1 and Theorem 2 will be applied to derive a feasible algorithm in Sect. 3.

3 Algorithm

Given a sample of directional functions $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n \subset \mathcal{F}$, a fixed number of clusters k and a fixed trimming level α:

1 *Initialize* B times: Each initialization starts from k randomly chosen initial centroids $\boldsymbol{m}_1^{(0)}, \ldots, \boldsymbol{m}_k^{(0)}$ (for instance, k randomly chosen $\boldsymbol{\theta}_i$ directional functions from our sample).

2 *Iterate:* Given centroids $\boldsymbol{m}_1^{(l-1)}, \ldots, \boldsymbol{m}_k^{(l-1)}$ at stage $l - 1$:

2.1 Let $d_{ij} = D(\boldsymbol{\theta}_i, \boldsymbol{m}_j^{(l-1)})$, $D_i = \min_{j=1,\ldots,k} d_{ij}$ and sort these values in $D_{(1)} \leq ... \leq D_{(n)}$.

2.2 Take $\mathcal{H}_j^{(l)} = \{i : d_{ij} = D_i \text{ and } D_i \leq D_{(\lceil n(1-\alpha) \rceil)}\}$ for $j = 1, ..., k$.

2.3 Update centroids $\boldsymbol{m}_1^{(l)}, \ldots, \boldsymbol{m}_k^{(l)}$, by considering the pointwise directional means:

$$t \mapsto \boldsymbol{m}_j^{(l)}(t) = \arg \left(\frac{\sum_{i \in \mathcal{H}_j^{(l)}} \exp(\mathrm{i} \cdot \boldsymbol{\theta}_i(t))}{\mathrm{card}\left(\mathcal{H}_j^{(l)}\right)} \right).$$

3 After L iterations of steps 2.1–2.4, we *compute* the value of the target function $\sum_{j=1}^k \sum_{i \in \mathcal{H}_j^{(L)}} D(\boldsymbol{\theta}_i, \boldsymbol{m}_j^{(L)})$ resulting from this random initialization.

4 Return as algorithm's output those partitions and templates yielding the smallest value in Step 3.

The theoretical justification of this algorithm follows from the application of Theorem 1 and Theorem 2, that guarantee the monotonically decrease of the target function in the iterative part of the algorithm. B random initializations are considered to avoid that the algorithm get stuck in local minima of the target function (3). A stopping rule can be also added to avoid unnecessary iterations if, after applying Step 2, the iterated solution does not change.

4 Algorithm with warping

When studying some processes it is usual to find some common patterns that occur at different speeds. One example can be found in some features of living beings (humans, animals, plants) of the same species where growth occur at different paces. Another example is the recognition of speech signals where the same word can be pronounced with varying speeds. It is in this last context where a family of techniques, known as dynamic time warping (DTW) algorithms, where introduced to deal with these different speeds (Sakoe and Chiba 1971). The global aim of these algorithms is to ensure that the varying speeds do not affect the similarity analysis of the curves and therefore allow the mentioned patterns to be detected. This is achieved through an appropriate time warping alignment function, ϕ, of the curves to be compared. A good reference that explains in more detail the theory of the DTW methodology applied here is Kruskal and Liberman (1983).

In the following we adapt the notation in Giorgino (2009) to our context. Let $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ two directional functions and $\phi = (\phi_1, \phi_2)$, where $\phi_1, \phi_2 : [0, 1] \to [0, 1]$ are two functions that warp the time for $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ respectively. In order to have consistent warpings, some constraints have to be imposed on ϕ. First, ϕ_1 and ϕ_2 must be non-decreasing continuous functions to ensure that time order is not reversed and to avoid time jumps. Also, starting and ending curve points must match, i.e., $\phi_1(0) = \phi_2(0) = 0$ and $\phi_1(1) = \phi_2(1) = 1$. In other words, we do not allow the beginning or end of any curve to be trimmed. In order to have some control over the local changes in time speed, we may also impose that $\phi_1(t), \phi_2(t) \in [t - \delta, t + \delta]$ for a preset value $\delta \in (0, 1]$, i.e., $\phi(t)$ must lie in a band around t.

Now, using the distance defined in (1), an accumulated distortion function d_ϕ is defined as

$$d_\phi(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = \int_0^1 d\big(\boldsymbol{\theta}_1(\phi_1(t)), \boldsymbol{\theta}_2(\phi_2(t))\big) m_\phi(t) M_\phi dt, \tag{5}$$

where $m_\phi(t)$ is a weighting function and M_ϕ the corresponding normalization constant, both to have comparable values for different choices of ϕ (see, e.g., Giorgino 2009, for more details). Then, the use of a DTW procedure to compare $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ requires to find the optimal value of $\overrightarrow{D}$ defined as:

$$\overrightarrow{D}(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = \inf_{\phi} d_{\phi}(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2),$$

under the assumed conditions on ϕ and where d in (5) corresponds to the extrinsic distance in $\mathbb{S}^1$.

To compute the value of $\overrightarrow{D}$ for two given curves $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ a numerical approximation of (5) is necessary, which we carry out through a discretisation in [0, 1] of $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$. If we consider a grid of equispaced nodes of size N in [0, 1] for both curves, then the computation of $\overrightarrow{D}$ can be carried out with DTW algorithms for discrete time series as those reviewed and described in Giorgino (2009), where the previous constraints on ϕ have a direct transpose. Although there is the possibility of estimating these ϕ functions, in our case we are only interested in computing the value of $\overrightarrow{D}$. Moreover, even though the computational complexity of the DTW algorithm for two time series of length N in the general case is $O(N^2)$, in our case, the computational complexity can be notably reduced when adding control over local changes in time speed.

The use of the DTW methodology is not essential in our proposal and could be replaced by other alternatives for curve registration (see, e.g., Marron et al. 2015,) and the same applies to the chosen constraints on the ϕ functions.

Our proposed algorithm follows similar lines as the algorithm introduced in Sangalli et al. (2010) but the DTW distance, $\overrightarrow{D}$, is used instead.

Given directional functions $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n \subset \mathcal{F}$, we search again for $\mathcal{H} = \mathcal{H}_1 \cup \ldots \cup \mathcal{H}_k \subset \{1, 2, ..., n\}$ with $\text{card}(\mathcal{H}) = \lceil n(1-\alpha) \rceil$ and for k templates $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_k \in \mathcal{F}$ minimizing

$$\sum_{j=1}^{k} \sum_{i \in \mathcal{H}_j} \overrightarrow{D}(\boldsymbol{\theta}_i, \boldsymbol{\xi}_j).$$

The k templates $\boldsymbol{\xi}_j$ may be seen as a kind of representative directional function for all the directional functions assigned (after warping) to cluster j. Notice that again a fraction α of directional functions with the (hopefully) most outlying behavior are trimmed.

A modified trimmed k-mean including warping can be given as:

1 *Initialize B* times: Each initialization starts from k randomly chosen initial templates $\boldsymbol{\xi}_1^{(0)}, \ldots, \boldsymbol{\xi}_k^{(0)}$ (as done in the algorithm presented in Sect. 3). k randomly chosen $\boldsymbol{\theta}_i$ directional functions from our sample can be chosen for this purpose.

2 *Iterate:* Given $\boldsymbol{\xi}_1^{(l-1)}, \ldots, \boldsymbol{\xi}_k^{(l-1)}$:

2.1 Let $d_{ij} = \overrightarrow{D}(\boldsymbol{\theta}_i, \boldsymbol{\xi}_j^{(l-1)})$, $D_i = \min_{j=1,\ldots,k} d_{ij}$ and $D_{(1)} \leq \ldots \leq D_{(n)}$.

2.2 Take $\mathcal{H}_j^{(l)} = \{i : d_{ij} = D_i \text{ and } D_i \leq D_{(\lceil n(1-\alpha)\rceil)}\}$ for $j = 1, \ldots, k$.

2.3 Let $\widetilde{\boldsymbol{\theta}_{ij}}$ be the directional function $\boldsymbol{\theta}_i$ after optimally warping it into the reference template $\boldsymbol{\xi}_j^{(l-1)}$, for $i = 1, \ldots, n$ and $j = 1, ..., k$.

2.4 Update templates as $\boldsymbol{\xi}_j^{(l)}$ being the pointwise directional mean of those $\widetilde{\boldsymbol{\theta}_{ij}}$'s warped directional functions with $i \in \mathcal{H}_j^{(l)}$.

3 After L iterations of steps 2.1–2.4, we *compute* the value of the target function

$$\sum_{j=1}^{k} \sum_{i \in \mathcal{H}_j^{(L)}} \overrightarrow{D}(\boldsymbol{\theta}_i, \boldsymbol{\xi}_j^{(L)}).$$

4 Return as the algorithm's output those partitions and templates yielding the smallest value in Step 3.

5 Choice of parameters

The correct choice of all the involved parameters, k (number of groups) and α (trimming level), is not an easy problem. The proper choice of k is a classical problem in Cluster Analysis and several proposals can be found in the literature trying to address it. The choice of α is an additional problem appearing now due to the trimming methodology adopted. Moreover, the choice of these two parameters should be done in an unified fashion because their effects are clearly interrelated. For instance, a high trimming level α could allow to entirely discard smaller clusters so that the total number of clusters k has to be decreased.

Sometimes the real data problem at hand provides some information on these two parameters, but in many others they are completely unknown and some guidance on their choice is welcome.

In this section, we review a simple approach introduced in García-Escudero et al. (2003), which is based on analyzing the decreasing pattern of the so-called trimmed k-variance functionals defined as

$$W_k : \alpha \mapsto W_k(\alpha) \text{ for } k = 1, 2, ...,$$

where $W_k(\alpha)$ is the minimum value attained in the minimization problem in (3) for fixed values of k and α. In fact, it is suggested the analysis of numerical second derivatives of these functionals. In order to approximate them, let us consider an equispaced grid of trimming levels $\{\alpha_1, \alpha_2, \dots, \alpha_L\} \subset [0, 1]$ with $\alpha_l = l/(L+1)$ and take

$$\widehat{W}_k''(\alpha_l) \approx \frac{W_k(\alpha_{l-h}) - 2W_k(\alpha_l) + W_k(\alpha_{l+h})}{(h/(L+1))^2},$$

defined for $l \in \{h+1, h+2, \dots, h-n\}$. We, thus, consider the numerical second derivative functionals as $W_k'' : \alpha_l \mapsto \widehat{W}_k''(\alpha_l)$, defined for $k = 1, 2, \dots$ and $l \in \{h+1, h+2, \dots, h-n\}$. The tuning parameter h controls the roughness of these numerical second derivative functionals, in such a way that they are more rough and data dependent when h is small. Notice also that high values of h make it impossible the determination of $W_k^{''}$ for some values of α_l close to 0.

We can say that K is a sensible choice for the number of clusters k if the associated numerical second derivative functionals are clearly different when $k < K$ but they almost coincide when $k \geq K$. In fact, detecting peaks in these functionals indicates that a higher number of clusters k is surely needed. Initial large and positive values for $\widehat{W}_k^{''}(\alpha_l)$ also indicates that a higher trimming level would be required as we are still probably trimming outlying observations with this α_l trimming level. A more detailed explanation for all these heuristic rules, together with simple justifications, can be found in García-Escudero et al. (2003).

6 Simulation study

We generate random sets of directional functions $\{\boldsymbol{\theta}_i\}_{i=1}^n$ such that

$$\boldsymbol{\theta}_i : [0,1] \to (\cos(\omega_i(t)), \sin(\omega_i(t))) \in \mathbb{S}^1 \tag{6}$$

where

$$\omega_i(t) = m_1(h_i(t)) \text{ with } m_1(t) = 2\pi \left(t + \tfrac{1}{3} e^{\frac{-(t-1/3)^2}{0.01}} \right) \text{ for } i = 1, \ldots, 20$$

and

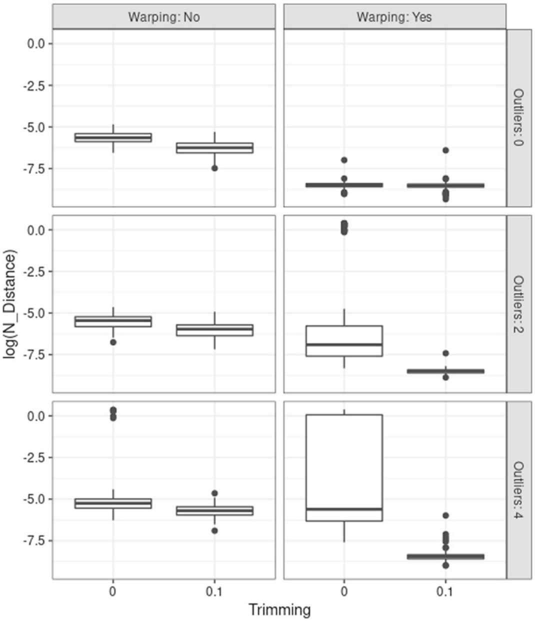
$$\omega_i(t) = m_2(h_i(t)) \text{ with } m_2(t) = 2\pi \left(t - \tfrac{1}{3} e^{\frac{-(t-2/3)^2}{0.01}} \right) \text{ for } i = 21, \ldots, 40.$$

In both cases, $h_i(\cdot)$ for $i = 1, \ldots, n$ are going to be random warping functions that are piecewise linearly defined and such that $h_i(0) = 0$, $h_i(1) = 1$ and $h_i(0.5) = a_i$ for a_i being randomly drawn from a normal distribution with mean equal to 0.5 and standard deviation equal to 0.07. We are so obtaining two clusters of (unaligned) directional functions. For applying the discretized version of our proposed methodology, we consider an equispaced grid of size 200 in the [0, 1] interval. To introduce contamination, we randomly replace 2 (5% contamination) or 4 (10% contamination) out of these 40 directional functions by directional functions defined as in (6) but with $\omega_i(t) = u_i + 2\pi t$ where u_i are randomly drawn from an uniform distribution in the $[0, 2\pi]$ interval.

Figure 1 summarizes the results obtained after applying the proposed methodology on 100 simulated data sets generated as explained above. Trimmed procedures are applied with $\alpha = 0.1$ (`Trimming: 0.1`) and compared with the untrimmed ones with $\alpha = 0$ (`Trimming: 0`). Warping can be considered with $\delta = 0.1$ (`Warping: Yes`) or not (`Warping: No`). The same 100 simulated data sets are applied for the comparison of the four different available approaches.

Each row in Fig. 1 shows the results for different numbers of outlying directional functions included (0, 2 and 4). The left column shows boxplots summarizing the

Fig. 1 Results of the simulation study in terms of the proportion of missclassified observations (left column) and in terms of the warped distance from the directional functions used to generate the two clusters and the output templates (right column). Different amount of outlying directional functions are considered in each row



performance of the procedures in terms of the proportion of wrongly classified directional functions among the non-contaminated ones. Additionally, the right column shows boxplots with $\log(N_{\text{Distance}})$ for

$$N_{\text{Distance}} = \min\left\{\overrightarrow{D}(\widetilde{\boldsymbol{\theta}}_1, \boldsymbol{\xi}_1) + \overrightarrow{D}(\widetilde{\boldsymbol{\theta}}_2, \boldsymbol{\xi}_2),\ \overrightarrow{D}(\widetilde{\boldsymbol{\theta}}_1, \boldsymbol{\xi}_2) + \overrightarrow{D}(\widetilde{\boldsymbol{\theta}}_2, \boldsymbol{\xi}_1)\right\},$$

where $\widetilde{\boldsymbol{\theta}}_1$ and $\widetilde{\boldsymbol{\theta}}_2$ are the target "reference" directional functions $\widetilde{\boldsymbol{\theta}}_j : [0,1] \to (\cos(m_j(t)), \sin(m_j(t))) \in \mathbb{S}^1$, for $j = 1, 2$, and $\boldsymbol{\xi}_1$ and $\boldsymbol{\xi}_2$ are the output templates obtained from the algorithms in each case, when $k = 2$. Recall that m_1 and m_2 are the functions used to generate the two clusters before considering the h_i warping functions. The N_{Distance} values are thus measuring how "close" the output templates are with respect to the target reference directional functions, after their proper alignment through $\overrightarrow{D}$.

We can see that even a small fraction of contaminating directional functions can create wrong assignment decisions that trimming is able to prevent once outlying directional functions can be removed. Figure 2c shows an example of the very bad performance of the proposed methodology allowing warping but without trimming. We can see how the two main clusters are artificially joined together and a small cluster made of few outliers is detected. Of course, the estimation of the target reference directional functions is harmfully affected.

Even in cases where no wrong assignments are obtained, it can be noticed that N_{Distance} seems to be reduced (with and without trimming) when warping is allowed in the algorithm. Given that we are allowing warping in $\overrightarrow{D}$ when computing N_{Distance}, it is easy to understand that methods allowing warping are going to provide better performance in that aspect. As can be seen in Fig. 2a, the simple pointwise directional mean (step 2.3 in the algorithm in Sect. 3) cannot provide $\boldsymbol{\xi}_j$ templates (wider blue lines) that correctly capture the target directional reference $\widetilde{\boldsymbol{\theta}}_j$ functions since the resulting $\boldsymbol{\xi}_j$ templates are clearly "oversmoothed" due to the non-alignment of the directional functions in each detected cluster. On the other hand, Fig. 2b shows that this oversmoothing phenomenon is corrected when considering a proper alignment before computing the pointwise directional means (step 2.4 in the algorithm in Sect. 4).

We might think that wrongly trimming some non-outlying directional functions could be detrimental, but we can see that the effect of a (slightly) greater trimming level than needed does not necessarily imply worse performance. In fact, smaller proportions of missclassified directional functions and smaller values of N_{Distance} are seen after trimming. This fact could be explained by how the most extremely unaligned curves are discarded as they are surely the trimmed ones.

7 Application to clustering of aircraft trajectories

The motivation of this application is framed within a research project named AIRPORTS: Airport Improvement Research On Processes & Operations of Runway, TMA & Surface leaded by Boeing Research and Technology Europe and devoted to analyze the efficiency of commercial flights when taking into account the aircraft trajectories actually flown. After a complex data intake and preprocessing procedure, some Key Performance Indexes (KPI) measuring important aspects as fuel consumption or polluting emissions, were computed. ADS-B (Automatic Dependent Surveillance-Broadcast) signals were considered to determine the real aircraft positions during their flights, after their proper integration with the planned routes from Eurocontrol. This implies a huge amount of information to be processed throughout more than 500 millions of ADS-B signals including, among others, information on the position (latitude, longitude and height) of each plane with a frequency that varies according to the receivers, but that in emission can be 2 times per second.

In a second phase, KPIs were constructed by comparing the real trajectories flown with respect to possible alternative synthetic trajectories that the plane could have flown (such as geodesics and geodesics based on the flight plan) which were generated by using a flight simulator owned by Boeing Research and Technology Europe. The different efficiency KPIs serve to detect which alternative trajectories would have been more efficient. Due to the huge number of trajectories and routes to be compared, Cluster Analysis methods applied to the trajectories were needed to carry out comparative studies between groups of trajectories with their respective KPIs. Moreover, in that clustering problem, there were numerous trajectories that can be considered as atypical and for which their automated detection was interesting. As we will see later, some of these atypical flights corresponds to operational deviations due to adverse weather conditions, congestion problems in the airspace, strikes, ... and also trajectories including ovals close to the destination to wait for the right moment to land.

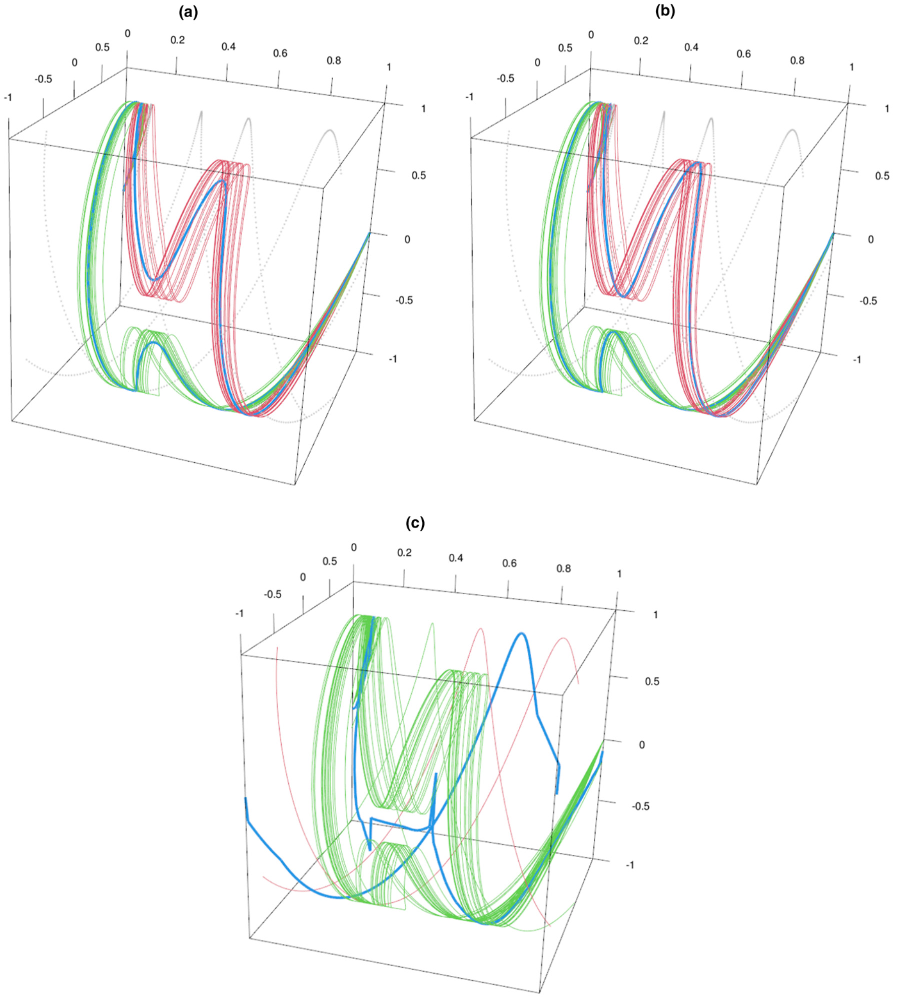


Fig. 2 Outputs of the algorithm for one of the samples in the simulation study. Green and red colors are used for showing functions belonging to the two clusters detected while trimmed directional functions appear in grey color. Blue color is used for representing the output templates. Plot **a** shows the result when warping and $\alpha = 0.1$ is considered. Plot **b** shows the result when no warping and $\alpha = 0.1$ is considered. Plot **c** shows the result when warping and $\alpha = 0$ is considered (color figure online)

In air navigation, heading is the horizontal angle between the direction of flight and magnetic north. It is common in aviation to characterize trajectories by measuring heading (data in $\mathbb{S}_1$), altitude (in meters or feet), and speed (in "mach" units). Using the evolution of these three features over time to group similar trajectories would be equivalent to looking for groups using the evolution of longitude, latitude and altitude (functional data in $\mathbb{R}^3$). To simplify this complex functional clustering problem, we focus exclusively on the evolution in time of only the heading after normalizing the flight times so that the time is restricted to the interval [0, 1] in such a way that this

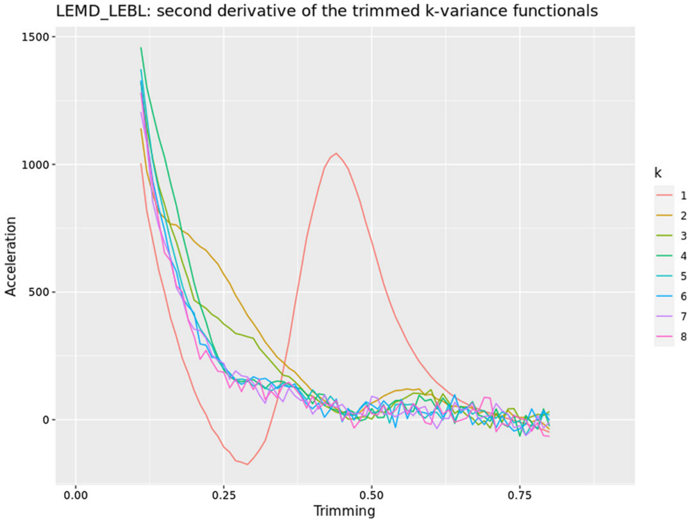


Fig. 3 Numerical second derivative of the trimmed k-variance functionals for the aircraft trajectories dataset

problem reduces to a clustering problem of directional functions in $\mathcal{F}$. This simplified problem, together with the need to avoid the damaging effect of atypical trajectories, led us to consider robust clustering for functional directional data as a very relevant problem to be addressed.

We will show a typical example for the problem addressed. In that example, a data set of $n = 3955$ aircraft trajectories corresponding to 6 months of flights from Madrid airport (LEMD) to Barcelona airport (LEBL) are clustered using the instant headings measured every second. Length of trajectories ranges from 2017 to 4494 seconds. After scaled to the interval [0, 1], we will apply the procedure described in Sect. 3 to the $\boldsymbol{\theta}_i(t)$ curves, $i = 1, \ldots, 3955$, representing the scaled heading of the aircrafts. The trajectories included in our dataset begin and end when the plane crosses the altitude threshold of 1000 meters.

To compute the values $\overrightarrow{D}$ described in Sect. 4 the size of the grid used in the discretisation has been $N = 2017$, i.e. the minimum length of the trajectories in the data set. Regarding the choice of the δ value to control local changes in time, although the main transformation of the time scale comes from the scaling to the interval [0, 1], we have selected $\delta = \frac{3}{2017}$, which means that locally we allow to advance or delay up to 3 steps in the grid. However, the choice of this δ value is not critical and identical or almost identical results are obtained for δ values that do not change much, e.g., up to 10 steps in this case. On the other hand, values above 100 steps in our case already produce deformations in the curves that are not very plausible.

In order to assess an appropriate value of k, as described in Sect. 5, a numerical approximation to the second derivative of the trimmed k-variance functionals is computed (see Fig. 3) with α_l trimming values $\{0.01, 0.02, ...\}$ and $h = 10$.

According to this figure, the minimum k that reaches similar numerical second derivative of the trimmed k-variance functionals with respect to the following ones is

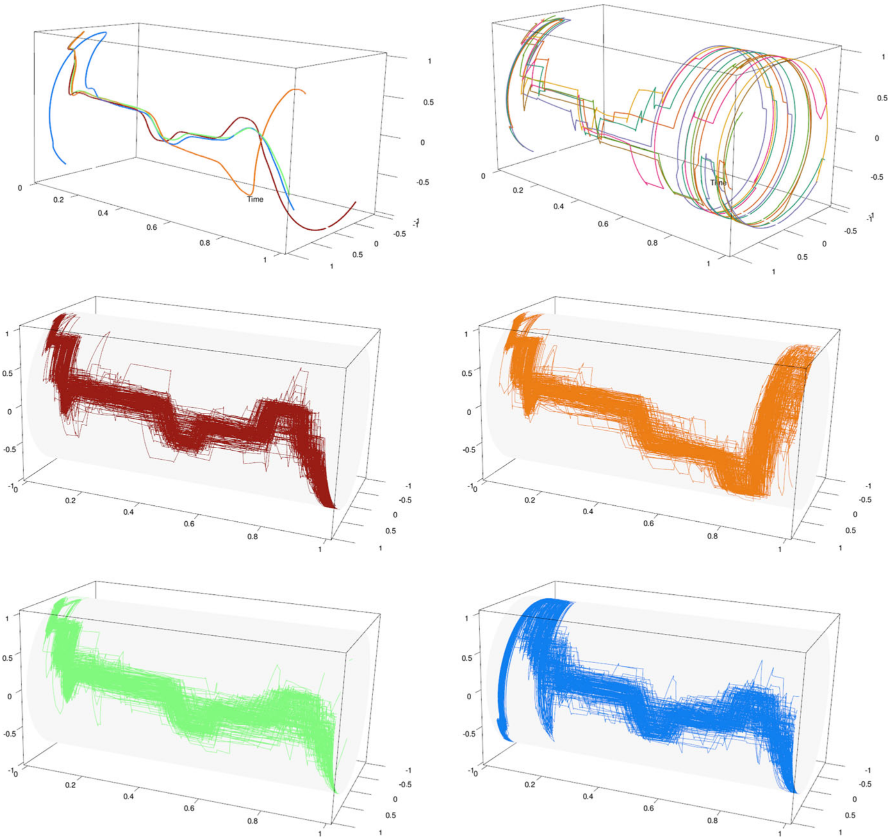


Fig. 4 (From left to right, top to bottom) Centroid trajectories for each cluster, some trimmed trajectories, and some trajectories in cluster 1–4 (dark red, orange, green and blue colors, respectively) (color figure online)

$k = 4$. We also observe that the acceleration (second derivative value) is positive and high for α values with $\alpha < 0.2$, but it is notable smaller and closer to 0 if we consider $\alpha > 0.2$. Then, a suitable value for α to remove outliers could be around $\alpha = 0.20$.

With these choices, $k = 4$ and $\alpha = 0.2$, we carry out the proposed methodology and obtain the clusters and centroids together with the trimmed trajectories. Figure 4 represents the $\boldsymbol{\theta}_i(t)$ trajectories in the cylinder $\mathcal{F}$ while Fig. 5 represent the trajectories in 2D.

Trimmed trajectories have been represented in 2D (white color) in the second panel of Fig. 5. It can be observed that most of these trajectories correspond to holding manoeuvres (oval courses), that take place in predeterminated places of the airspace just before the arrival to destination; and some other strange trajectories. These holding patterns can be observed at the cylinder in the second plot of Fig. 4 as trajectories that turns around one or more times.

Clusters can be mainly described in terms of the departing and arrival runway direction used, but not only. Cluster 4 (blue color) is composed by trajectories that

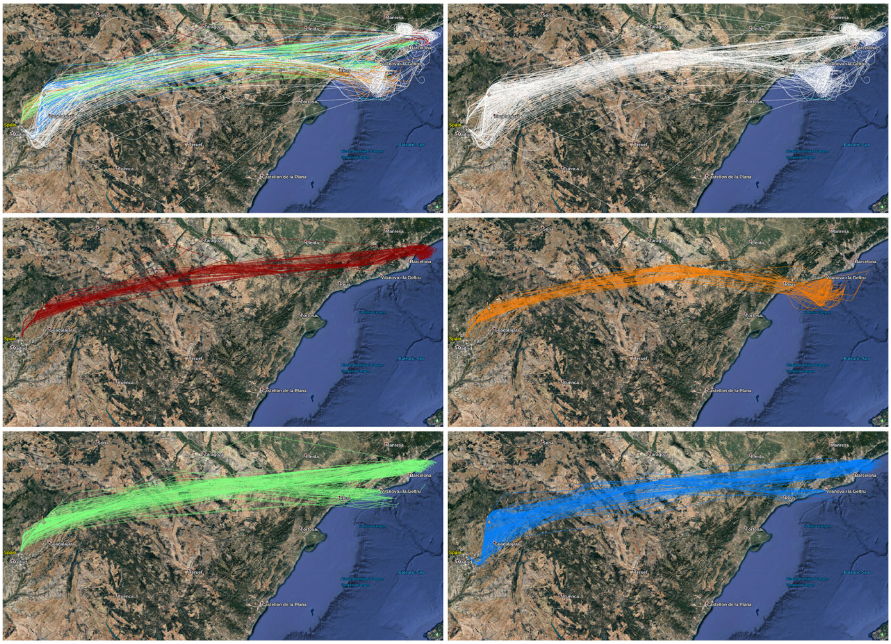


Fig. 5 (From left to right, top to bottom) All 2D trajectories, trimmed trajectories (white), trajectories in cluster 1–4 (dark red, orange, green and blue colors, respectively) (color figure online)

depart from the south of the airport and approaches destination both from the northeast and southwest. Those from the northeast are in the majority and this is reflected in the corresponding centroid (blue color) in the first plot of Fig. 4. On the opposite, clusters 1–3 are composed by trajectories which depart from the north of LEMD, and the main differences among them are in the way of approaching LEBL. Cluster 1 (dark red) is composed by trajectories that approach LEBL from the northeast and the last section is characterized by a sharp turn towards the airport. Cluster 2 (orange) is composed by trajectories that approach LEBL from the southeast but before the last section of the path they open up to the sea (to the east), in a sharp turn. Finally, cluster 3 (green) is composed by trajectories that approach LEBL both from the northeast -but with a smoother (than those in cluster 1) turn-, and from the southwest -but without the turn observed in Cluster 2-.

One could think of a registration process (alignment or “warping”) that allows detecting representative routes described by their headings regardless of their speeds. This will serve to simplify future analyses by grouping flights close to these representative routes. The DTW method makes it easy to control the maximum degree of “time lag” allowed.

Figure 6 shows the output of applying the algorithm with warping of Sect. 4 to the previous dataset, using $k = 4$ and $\alpha = 0.20$. The trajectories drawn in the four plots show very slight differences with the corresponding ones in Fig. 4 indicating that in this case the scaling to the interval [0, 1] is sufficient to correct the differences in flight durations and speeds.

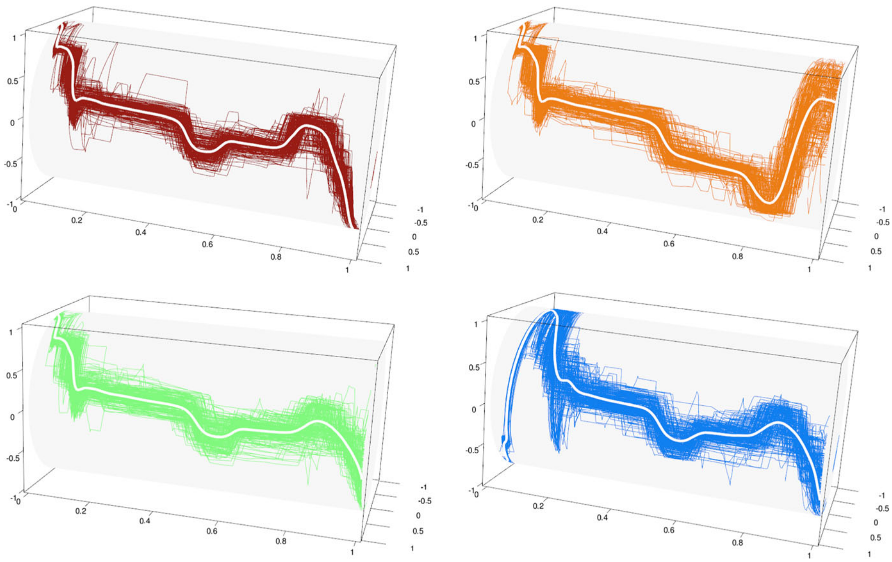


Fig. 6 Output of the algorithm with warping applied to the aircraft trajectories example ($k = 4$, $\alpha = 0.2$). From left to right, top to bottom, some trajectories in cluster 1–4 (dark red, orange, green and blue colors, respectively). White color curves represent the template routes, $\boldsymbol{\xi}_j$, for each group $j = 1, \ldots, 4$ (color figure online)

The use of alternative registration procedures like those in Srivastava and Klassen (2016) could be another approach to follow in this type of problem.

8 Conclusions and further directions

A new methodology for robust clustering directional functional data has been introduced and a feasible algorithm has been proposed and justified. Robustness against outlying curves is pursued by allowing that a fixed fraction α of curves, hopefully the most outlying ones, are left unassigned or trimmed. The procedure is extended to allow for time warpings in the directional functions. An application to group aircraft trajectories is presented to illustrate the interest of the proposed methodology.

This work reveals many potential lines of work to consider in the future. For instance, it is interesting to study the possibility of carrying out a feasible dimensionality reduction in a way that can alleviate the computational cost or adequately handle the periodicity in the observed curves. Providing more automated ways of determining the parameters k and α in this particular problem is also an interesting line of research. As with other methods that follow a k-means approach, the procedure ideally assumes that the underlying clusters have the same spread/variation compared to other clusters. To address this problem, it might make sense to consider trimmed versions of mixtures of von Mises-Fisher distributions (Banerjee et al. 2003) in a way analogous to how TCLUST (García-Escudero et al. 2008) generalizes the trimmed k-means. Finally, trimming entire curves can be very extreme if outlying measurements only occur at a few particular time points for that curve. It would be useful to develop procedures that are capable of discarding only the outlying measurements in each curve and keeping the valuable information in non-outlying measurements.

Acknowledgements We would like to thank Boeing Research and Technology Europe (BRT&E) for the access to the dataset used in Sect. 7. The authors also thank the associate editor and two anonymous referees for their constructive comments, which have been very helpful in improving the manuscript.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature.

References

Banerjee A, Dhillon I, Ghosh J, Sra S (2003) Generative model-based clustering of directional data. In: Proceedings of the ninth ACM SIGKDD international conference on knowledge discovery and data mining. pp. 19–28

Cuesta-Albertos JA, Fraiman R (2007) Impartial trimmed k-means for functional data. Comput Stat Data Anal 51:4864–4877

Cuesta-Albertos JA, Gordaliza A, Matrán C (1997) Trimmed k-means: an attempt to robustify quantizers. Ann Stat 25:553–576

Ferraty F, Vieu P (2006) Nonparametric functional data analysis. Springer Series in Statistics, Springer, New York

García-Escudero L, Gordaliza A, Matrán C, Mayo-Iscar A (2008) A general trimming approach to robust cluster analysis. Ann Stat 36:1324–1345

García-Escudero L, Gordaliza A, Matrán C, Mayo-Iscar A (2010) A review of robust clustering methods. Adv Data Anal Classif 4:89–109

García-Escudero L, Gordaliza A, Matrán C, Mayo-Iscar A, Hennig C (2015) Robustness and Outliers, chapter 29, Chapman & Hall/CRC handbooks of modern statistical methods. Taylor & Francis. pp. 653–678

García-Escudero LA, Gordaliza A (1999) Robustness properties of k means and trimmed k means. J Am Stat Assoc 94:956–969

García-Escudero LA, Gordaliza A (2005) A proposal for robust curve clustering. J Classif 22:185–201

García-Escudero LA, Gordaliza A, Matrán C (2003) Trimming tools in exploratory data analysis. J Comput Gr Stat 12:434–449

Giorgino T (2009) Computing and visualizing dynamic time warping alignments in R: the dtw package. J Stat Softw 31(7):1–24

Gordaliza A (1991) Best approximations to random variables based on trimming procedures. J Approx Theor 64:162–180

Hitchcock D, Greenwood M (2015) Robustness and Outliers, chapter 13, Chapman & Hall/CRC Handbooks of modern statistical methods. Taylor & Francis. pp. 265–287

Jacques J, Preda C (2014) Functional data clustering: a survey. Adv Data Anal Classif 8:231–255

Kruskal JB, Liberman M (1983) The symmetric time-warping problem: from continuous to discrete. In: Sankoff D, Kruskal JB (eds) Time Warps, String Edits, and Macromolecules: the Theory and Practice of Sequence Comparison. Addison-Wesley Publishing Company, pp. 125–161

Ley C, Verdebout T (2017) Modern directional statistics. CRC Press

Mardia KV, Jupp PE (2009) Directional statistics. Wiley, New York

Marron JS, Ramsay JO, Sangalli LM, Srivastava A (2015) Functional data analysis of amplitude and phase variation. Stat Sci 30:468–484

Ramsay JO, Silverman BW (2005) Functional data analysis. Springer Series in Statistics, Springer, New York

Ritter G (2015) Robust cluster analysis and variable selection, volume 137 of Monographs on statistics and applied probability. CRC Press, Boca Raton

Rivera-García D, García-Escudero LA, Mayo-Iscar A, Ortega J (2019) Robust clustering for functional data based on trimming and constraints. Adv Data Anal Classif 13:201–225

Rousseeuw PJ (1984) Least median of squares regression. J Am stat Assoc 79:871–880

Sakoe H, Chiba S (1971) A dynamic programming approach to continuous speech recognition. In: Proceedings of the seventh international congress on acoustics

Sangalli LM, Secchi P, Vantini S, Vitelli V (2010) K-mean alignment for curve clustering. Comput Stat Data Anal 54(5):1219–1233

Srivastava A, Klassen EP (2016) Functional and shape data analysis. Springer, New York

Yassouridis C, Leisch F (2017) Benchmarking different clustering algorithms on functional data. Adv Data Anal Classif 11:467–492

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