Noname manuscript No.
(will be inserted by the editor)

Robust archetypoids for anomaly detection in big functional data

G. Vinué · I. Epifanio

Received: date / Accepted: date

Abstract Archetypoid analysis (ADA) has proven to be a successful unsupervised statistical technique to identify extreme observations in the periphery of the data cloud, both in classical multivariate data and functional data. However, two questions remain open in this field: the use of ADA for outlier detection and its scalability. We propose to use robust functional archetypoids and adjusted boxplot to pinpoint functional outliers. Furthermore, we present a new archetypoid algorithm for obtaining results from large data sets in reasonable time. Functional time series are occurring in many practical problems, so this paper focuses on functional data settings. The new algorithm for detecting functional anomalies, called CRO-FADALARA, can be used with both univariate and multivariate curves. Our proposal for outlier detection is compared with all the state-of-the-art methods in a controlled study, showing a good performance. Furthermore, CRO-FADALARA is applied to two large time series data sets, where outliers curves are discussed and the reduction in computational time is clearly stated. A third case study with a small ECG data set is discussed, given its importance in functional data scenarios. All data, R code and a new R package are freely available.

Keywords Anomaly detection · Functional data analysis · Archetypal analysis · Big data · R package

G. Vinué✉
KU Leuven, Department of Computer Science
Celestijnenlaan 200A box 2402, 3001 Leuven, Belgium
E-mail: guillermovinue@gmail.com

I. Epifanio
Dept. Matemàtiques and IF
Campus del Riu Sec. Universitat Jaume I, 12071 Castelló, Spain
E-mail: epifanio@mat.uji.es

1 Introduction

Archetypoid analysis (ADA) (Vinué et al., 2015) is an extension of archetype analysis (AA) (Cutler and Breiman, 1994) and has become a useful unsupervised statistical technique for finding extreme observations. Evidence of this is the number of successful applications, both with classical continuous multivariate data, functional data and other kind of data (binary data, shapes, etc.), such as ergonomy and anthropometry (Vinué et al., 2015; Epifanio et al., 2018; Alcacer et al., 2020), weather temperatures and the study of human development around the world (Epifanio, 2016; Epifanio et al., 2020), hyperspectral imagery (Sun et al., 2017; Cabero and Epifanio, 2019), sports (Vinué and Epifanio, 2017, 2019), social sciences (Cabero and Epifanio, 2020), financial time series (Moliner and Epifanio, 2019) and water networks (Millán-Roures et al., 2018). However, there are still two open questions in archetypoids theory: exploiting its use for anomaly detection and scalability. We address both in this paper.

In the current era of big data, a lot of the data that is generated every day is time series in nature. A time series is a sequence of data points indexed in time order. Time series analysis and functional data analysis (FDA) are two different statistical methodologies to approach time series, being time series analysis a model-based design, whereas FDA is not.

The idea of FDA is to consider observed data as single functional entities, rather than as a sequence of individual observations. The term functional refers to the intrinsic structure of the observed data, rather than to their explicit form. Then, in FDA each curve is a datum. Assuming that a datum for replication i arrives as a set of discrete measured values, $y_{i1}, \dots, y_{in}$, the first thing to do is to convert these values to a function x_i with values $x_i(t)$ computable for any value of the parameter t. The conversion from discrete data to underlying functions involves smoothing, so that a pair of adjacent data values, y_j and y_{j+1} are linked together to some extent and are not too different from each other. By smooth, we mean that function x_i has one or more derivatives, denoted by Dx_i, D^2x_i, and so on, so that $D^m x_i$ refers to the derivative of order m, and $D^m x_i(t)$ is the value of that derivative at argument t. Additionally, FDA works well in cases where time series contain missing data or where points in time are not successive equally spaced. Thus, FDA offers a wider range of possibilities to analyze time-dependent data.

In view of the above considerations, we are going to use FDA as our analytic tool for time-dependent data. There are currently two references that excel in providing an introduction to FDA: Ramsay and Silverman (2005) and Ramsay et al. (2009). The R package **roahd** also proposes an advanced analysis of high dimensional functional data using robust nonparametric statistics (Tarabelloni et al., 2018).

As with any other field of statistics, outliers can seriously impact the modeling of functional data, leading to incorrect conclusions. An outlier is defined as the curve generated by a stochastic process different from the rest of the curves, which are assumed to be identically distributed (Febrero et al., 2008).

A taxonomy of different types of functional outliers was set up in Hubert et al. (2015), in terms of defining the main characteristics of each one. Arribas and Romo (2014) also provided an interesting overview of the outliers types in multivariate functional analysis, focusing particularly on shape outliers. Curves that exhibit outlying behavior during a very short time period are called isolated outliers. On the contrary, a long time outlying activity indicates persistent outliers. Three types of persistent outliers can be distinguished: (i) shift/magnitude outliers, which are identical to the other curves, but are moved away; (ii) amplitude outliers, which are identical to the other curves, but their scale differs; (iii) shape outliers, which are curves whose shape differs from the majority (although they might not stand out at any time point).

In order to structure a large multivariate or functional data set, clustering is the traditional way for grouping observations, such that similar points are gathered together and separated from dissimilar ones. If the goal is investigating the typical data points, the average is very likely to be a good data representation. However, the average does not serve well if the goal is to identify a set of contrasting categories. Archetypes, the pure types of data, are extreme points in the periphery of the data cloud. Therefore, they are the correct form for obtaining a contrasting categorization. When archetypes are real observations they are known as archetypoids. The main feature of archetypoid analysis is that each datum is expressed as a mixture of archetypoids. Therefore, it allows us to identify not only extreme observations, but also the approximation of other observations according to the archetypoids. In both multivariate and functional scenarios, an outlier is an observation that lies outside the different patterns observed in data. Because archetypes and archetypoids are derived from extreme observations, it is very likely that some of them would correspond with any outlier present in the data set, if the residuals are not moderated. In other words, they are sensitive to anomalies. In order to avoid this influence of outliers, robust procedures must be considered. Robust analysis is the field of statistics that allows us to avoid the outlier effects. Its basic principle is to fit the majority of the data, after which anomalies are identified as those points which possess large residuals from the robust solution (Rousseeuw and Leroy, 1987).

A first attempt to formulate the robust version of archetypes was described in Eugster and Leisch (2011). A further improvement has been presented in Moliner and Epifanio (2019), where the new objective function is the bisquare family of loss functions defined from $\mathbb{R}^+$ to $\mathbb{R}$. This second attempt is showing better performance in terms of obtaining a more robust solution. However, it does not explain how to pinpoint outliers. An iterative procedure for identifying outlier functions based on classical archetypoids was presented in Millán-Roures et al. (2018). This approach is not based on robust features, so it suffers from a high false positive rate. This was highlighted in the simulation study discussed in Millán-Roures et al. (2018). As a consequence of the above, the question of using archetypoids for anomaly detection using robust features remains open. To fill this gap, we propose a new procedure that combines the use of robust archetypoids and the adjusted boxplot for skewed distributions.

The resulting algorithm is called CRO-FADA in the functional setting. This is a non-iterative algorithm, unlike Millán-Roures et al. (2018).

The classical archetypoid algorithm still has an important weakness: poor scalability (and therefore low efficiency). This is because the coefficients that represent how much each archetypoid contributes to the approximation of each individual must be recalculated every time that a new set of archetypoids is identified. To address these issues, we present a new algorithm, called CRO-FADALARA in the functional setting, based on a sampling strategy. The use of sampling enables any method to deal with large data sets well. A second version of CRO-FADALARA using parallel computing is also presented, with the aim of further decreasing the computational time by using the full power of the computer. As a final remark, most of the approaches developed so far for functional outlier detection were restricted to univariate curves. Arribas and Romo (2014) and Hubert et al. (2015) were the first two proposals to identify multivariate functional outliers. Arribas and Romo (2014) combine the Modified Band Depth and the Modified Epigraph Index to create a graphical representation called outliergram. This methodology is mainly devoted to detect shape outliers, but magnitude outliers can also appear at the bottom corners of the plot. On the other hand, Hubert et al. (2015) also use depth functions and distance measures derived from them. CRO-FADALARA accepts as many variables as desired. We have applied CRO-FADALARA to three different time series databases. For the first two cases, in order to inspect the results, we have created two interactive web applications, one per data set, using the R package **shiny** (Chang et al., 2017).

The main novelties and innovations of this work consist of:

- proposing a new method based on robust functional ADA and the adjusted boxplot for skewed distributions to detect functional outliers: CRO-FADA.
- showing the good performance of CRO-FADA in comparison with functional benchmark methods in a simulation study.
- presenting a new archetypoid algorithm, CRO-FADALARA, which is able to scale to large functional data sets (not only with univariate but also with multivariate). The idea also serves for scaling ADA for multivariate classical data.
- showing the dramatic reduction in computational time of CRO-FADALARA with respect to CRO-FADA.
- showing how CRO-FADALARA can be used in a common case study for anomaly detection and how its results can be inspected with a web application.
- presenting the new R package **adamethods** `https://cran.r-project.org/package=adamethods`, that includes all the archetypoid-based algorithms and is on the Comprehensive R Archive Network (R Core Team, 2018).

This paper adheres to the best practices of reproducible research, by making freely available all data and R code used (including the web applications) at `https://www.uv.es/vivigui/softw/code_and_data_crofadalara.zip`. They can also be found in the supplementary material of the paper. The rest of the paper is organized as follows. Section 2 introduces all the previous work related to both classical, functional and robust archetypoids, as well as to functional outlier detection methods. The proposed method is introduced in Section 3. In Section 4 our main results are exposed. Section 5 ends the paper with some conclusions.

2 Related work

2.1 ADA for classical multivariate data

In ADA, archetypes correspond to real observed cases (the so-called archetypoids). Let $\mathbf{X}$ be an $n \times p$ matrix of real numbers representing a multivariate data set with n observations and p variables. For a given number of archetypoids g, the goal of ADA is to obtain the $n \times g$ coefficient matrix α and the $g \times n$ matrix β which minimize the following residual sum of squares (RSS):

$$RSS = \sum_{i=1}^{n} \|\mathbf{x}_i - \sum_{j=1}^{g} \alpha_{ij}\mathbf{z}_j\|^2 = \sum_{i=1}^{n} \|\mathbf{x}_i - \sum_{j=1}^{g} \alpha_{ij} \sum_{l=1}^{n} \beta_{jl}\mathbf{x}_l\|^2, \tag{1}$$

under the constraints

1) $\sum_{j=1}^{g} \alpha_{ij} = 1$ with $\alpha_{ij} \geq 0$ and $i = 1, \ldots, n$ and
2) $\sum_{l=1}^{n} \beta_{jl} = 1$ with $\beta_{jl} \in \{0, 1\}$ and $j = 1, \ldots, g$ i.e., $\beta_{jl} = 1$ for one and only one l and $\beta_{jl} = 0$ otherwise.

On the one hand, constraint 1) implies that the predictors of $\mathbf{x}_i$ are convex combinations of the collection of archetypoids $\mathbf{z}_j$, $\hat{\mathbf{x}}_i = \sum_{j=1}^{g} \alpha_{ij}\mathbf{z}_j$. The α coefficients represent how much each archetypoid contributes to the approximation of each observation. On the other hand, constraint 2) means that archetypoids are real data points. The $g \times p$ matrix $\mathbf{Z} = \beta\mathbf{X}$ characterizes the archetypal patterns in the data. Archetypoids are computed with the R package **Anthropometry** (Vinué, 2017).

The archetypoid algorithm has two phases: a BUILD phase and a SWAP phase. In the BUILD step, an initial set of archetypoids is determined. In the algorithm described in Vinué et al. (2015), there were three options to compute this set from the results of the AA algorithm defined by Cutler and Breiman (1994). From our practical experiments, the three options give very close or even the same results. Our goal in this paper is to speed up computations to make ADA feasible for large data. Therefore, only one option is going to be used now. This consists in computing the Euclidean distance between the g archetypes and all the individuals, and choosing the nearest ones. Finally, the SWAP phase tries to improve the current set of archetypoids by exchanging selected cases for unselected cases and by checking whether or not these replacements reduce the RSS objective function. Note that in order to avoid local minima, the AA algorithm is started several times with different initial archetypes, which are randomly selected. This strategy is also useful for avoiding local minima derived from the choice of a redescending M-estimator. In any case, the problem should not be very severe in view of the good performance obtained using only a small number of restartings.

2.2 ADA for functional data

In FDA, the values of the p variables of the classical multivariate data become function values with a continuous index t. The database here is made up of the set of $\{x_1(t), \dots, x_n(t)\}$ univariate functions with $t \in [a, b]$. It is always assumed that the functions belong to a Hilbert space, fulfill reasonable smoothness conditions and are square-integrable functions on $[a, b]$. The transition of ADA to deal with functional data was explained in Epifanio (2016). In functional archetypoid analysis (FADA), vectors are replaced by functions. Similarly to ADA, the goal of FADA is to find g archetypoid functions $z_j(t)$ in the sample, in such a way that the other sampled functions can be approximated through the mixtures of these archetypoids. The interpretation of the matrices α and β remains the same as in ADA. In Eq. 1, the vector norm is replaced by the L^2-functional norm, $||f||^2 =< f, f >= \int_a^b f(t)^2 d(t)$.

In practice, the functions are not observed continuously, but rather in a finite set of time points. A first approach could be to discretize the functions to a grid of p equally spaced values from a to b and to apply ADA to the $n \times p$ matrix $\mathbf{X}$. However, this is not computationally feasible, especially in the case of dealing with large data sets (Epifanio, 2016). The most popular alternative is to use basis function expansions. A basis function system is a set of functions B_h that are linearly independent of each other. Each function $x_i(t)$ is constructed as a linear combination of these basis functions, $x_i(t) = \sum_{h=1}^{p} b_i^h B_h(t) = \mathbf{b'}_i \mathbf{B}$, where $'$ stands for transpose, $\mathbf{b}_i$ is the vector of length p of the coefficients and $\mathbf{B}$ is the functional vector whose elements are the basis functions. The residual sum of squares (RSS) of the FADA problem is computed now as follows:

$$RSS = \sum_{i=1}^{n} \|x_i - \sum_{j=1}^{g} \alpha_{ij} z_j\|^2 = \sum_{i=1}^{n} \|x_i - \sum_{j=1}^{g} \alpha_{ij} \sum_{l=1}^{n} \beta_{jl} x_l\|^2 = \sum_{i=1}^{n} \mathbf{a'}_i \mathbf{W} \mathbf{a}_i, \quad (2)$$

where $\mathbf{a'}_i = \mathbf{b'}_i - \sum_{j=1}^{g} \alpha_{ij} \sum_{l=1}^{n} \beta_{jl} \mathbf{b'}_l$ and $\mathbf{W}$ is the order m symmetric matrix with the inner products of the pairs of basis functions, $w_{m_1,m_2} = \int B_{m_1} B_{m_2}$.

If the functions are expressed in an orthonormal basis, $\mathbf{W}$ is the order p identity matrix and functional archetypoids can be obtained as in the classical multivariate case simply by applying ADA to the basis coefficients. Otherwise, numerical integration has to be used to compute $\mathbf{W}$ as a first step.

Ideally, basis functions should be able to reproduce any feature of interest of the original data. This approach has the advantage of being more computationally efficient, because the number of coefficients of the basis functions is usually smaller than the original number of time points. This is a common procedure in FDA (Ramsay and Silverman, 2005, Section 4.5).

Real data is usually expressed in several dimensions. This is also the case of functional data. The key point in multivariate functional data is to compute an inner product between multivariate functions. The simplest definition is to sum the inner products of the multivariate functions. Thus, the squared norm of a P-variate function is the sum of the squared norms of the P components. This means that FADA for P-variate functions is equivalent to P independent FADA, with shared matrices α and β.

In the interest of illustration, the bivariate case will be defined. Let $\mathbf{f}_i(t) = (x_i(t), y_i(t))$ be a bivariate function. Its squared norm is $||\mathbf{f}||^2 = \int_a^b x_i(t)^2 dt + \int_a^b y_i(t)^2 dt$. In addition, let $\mathbf{b^x}_i$ and $\mathbf{b^y}_i$ be the vectors of length p of the coefficients for x_i and y_i for the basis functions B_h. The residual sum of squares is computed in this case as follows:

$$\begin{aligned} RSS &= \sum_{i=1}^{n} \|\mathbf{f}_i - \sum_{j=1}^{g} \alpha_{ij} \mathbf{z}_j\|^2 = \sum_{i=1}^{n} \|\mathbf{f}_i - \sum_{j=1}^{g} \alpha_{ij} \sum_{l=1}^{n} \beta_{jl} \mathbf{f}_l\|^2 \\ &= \sum_{i=1}^{n} \|x_i - \sum_{j=1}^{g} \alpha_{ij} \sum_{l=1}^{n} \beta_{jl} x_l\|^2 + \sum_{i=1}^{n} \|y_i - \sum_{j=1}^{g} \alpha_{ij} \sum_{l=1}^{n} \beta_{jl} y_l\|^2 \\ &= \sum_{i=1}^{n} \mathbf{a^x}'_i \mathbf{W} \mathbf{a^x}_i + \sum_{i=1}^{n} \mathbf{a^y}'_i \mathbf{W} \mathbf{a^y}_i, \end{aligned} \tag{3}$$

where $\mathbf{a^x}'_i = \mathbf{b^x}'_i - \sum_{j=1}^{g} \alpha_{ij} \sum_{l=1}^{n} \beta_{jl} \mathbf{b^x}'_l$ and $\mathbf{a^y}'_i = \mathbf{b^y}'_i - \sum_{j=1}^{g} \alpha_{ij} \sum_{l=1}^{n} \beta_{jl} \mathbf{b^y}'_l$. The predictors of $\mathbf{f}_i$ are convex combinations of the collection of archetypoids $\mathbf{z}_j$, $\hat{\mathbf{f}}_i = \sum_{j=1}^{g} \alpha_{ij} \mathbf{z}_j$. The union of $\mathbf{b^x}_i$ and $\mathbf{b^y}_i$ results in the set of data observations. Again, if the basis functions are orthonormal, FADA reduces to apply standard ADA for the $n \times 2p$ coefficient matrix composed by joining the coefficient matrix for x and y components.

2.3 Robust ADA and FADA

As shown in previous sections, RSS is formulated as the sum of the squared (vectorial or functional) norm of the residuals, r_i $(i = 1, \ldots, n)$, namely,

$RSS = \sum_{i=1}^{n} \|\mathbf{r}_i\|^2 = \sum_{i=1}^{n} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2$ in the classical multivariate case, $RSS = \sum_{i=1}^{n} \|r_i(\cdot)\|^2 = \sum_{i=1}^{n} \|x_i(\cdot) - \hat{x}_i(\cdot)\|^2$ in the univariate functional case and $RSS = \sum_{i=1}^{n} \|\mathbf{r}_i(\cdot)\|^2 = \sum_{i=1}^{n} \|\mathbf{f}_i(\cdot) - \hat{\mathbf{f}}_i(\cdot)\|^2$ in the multivariate functional case. Note that the residuals r_i are vectors ($\mathbf{r}_i$) in the classical multivariate case, functions ($r_i(\cdot)$) in the univariate functional case and vector-valued functions ($\mathbf{r}_i(\cdot)$) in the multivariate functional case.

This formulation based on the least squared loss function is not robust because large residuals have large effects. M-estimators are often used as a robust replacement of the least squared loss. They aim to decrease the effect of outliers by replacing the squared residuals by a less rapidly increasing loss function. Sinova et al. (2018) proposed M-estimators in the functional setting as:

$$\hat{\theta}^{\mathbb{H}}(\cdot) = \text{argmin}_{s \in \mathbb{H}} \sum_{i=1}^{n} \rho_c(||x_i(\cdot) - s(\cdot)||_{\mathbb{H}}) = \rho_c(||r_i(\cdot)||_{\mathbb{H}}), \tag{4}$$

where ρ_c is a loss function and $|| \cdot ||_{\mathbb{H}}$ is a norm for a Hilbert space $\mathbb{H}$. In Sinova et al. (2018), the conditions of the loss function ρ_c for functional M-estimators are fully described. The most remarkable aspects are: (i) ρ_c is a continuous and non-decreasing function, whose domain definition is from $\mathbb{R}^+$ to $\mathbb{R}$; (ii) $\rho_c(0) = 0$; (iii) $\rho_c(x)/x$ should tend towards zero, when x tends towards zero; (iv) ρ_c should be differentiable, and both ρ' and $\phi(x) = \rho'(x)/x$ should be continuous and bounded, where we assume that $\phi(0) := lim_{x \to 0}$ $\rho'(x)/x$ exists and is finite. The standard least squared loss function $\rho(x) = x^2$ does not satisfy this last condition (ρ' is not bounded). In Sinova et al. (2018), we can also find details about properties of functional M-estimators, such as their consistency and robustness by means of their breakdown point and their influence function.

For obtaining a robust estimation of functional archetypoids, we use the same definition of M-estimators as in Sinova et al. (2018). Specifically, RSS in Eqs. (1), (2) and (3) is replaced by $\sum_{i=1}^{n} \rho_c(||r_i||)$, where $|| \cdot ||$ respectively denotes the Euclidean norm for vectors, the L^2-norm for univariate functions and the corresponding norm for P-variate functions. For example, the robust estimation of functional archetypoids for the univariate functional case would be exactly as Eq. (4), where $s(\cdot) = \sum_{j=1}^{g} \alpha_{ij} \sum_{l=1}^{n} \beta_{jl} x(\cdot)_l$.

We choose the Tukey biweight or bisquare family of loss function (Beaton and Tukey, 1974), following the ideas of Moliner and Epifanio (2019). This loss function has a very good performance with extreme outliers. Therefore, $\rho_c(||r_i||)$ is given by:

$$\rho_c(||r_i||) = \begin{cases} c^2/6 \times ((1 - (1 - ||r_i||^2/c^2)^3)) & \text{if } 0 \leq ||r_i|| \leq c \\ c^2/6 & \text{if } c < ||r_i||. \end{cases} \tag{5}$$

The tuning parameter c will be obtained from the calculation of a user-given quantile of the vector containing the non-zero residual norms. As explained in Sinova et al. (2018), the M-estimators are not scale equivariant in general. As a consequence, the choice of c should be data-dependent. Other approaches have used the Huber loss function as an alternative to bisquare (Sun et al., 2017; Chen et al., 2014). However, it is important to emphasize that the bisquare family deals better with extreme outliers, since the norm of residuals larger than c contribute the same to the loss, which is not the case with the Huber family.

The acronyms for robust ADA and robust FADA are R-ADA and R-FADA, respectively.

2.4 Outlier detection in skewed distributions

According to the robust statistics theory, outliers will be the points showing large residuals. However, the main question here is how to decide if a residual is "large". In Least Median of Squares (LMS) regression, residuals are compared with a robust estimation of the standard deviation (Rousseeuw and Leroy, 1987). However, ADA and FADA are distribution free, so an estimate of the standard deviation of the norm of the residuals is hard to find. In addition, the distribution of the norm of ADA and FADA residuals has the peculiarity to be highly skewed. We have considered two alternatives to cope with this issue:

(i) Nonparametric (i.e., distribution-free) tolerance intervals (Young, 2010). Tolerance intervals provide limits within which at least a certain proportion of the population falls, with a given confidence level. Therefore, they can be used to find uncommon values.

(ii) The adjusted boxplot for skewed distributions (Hubert and Vandervieren, 2008), which is detailed next.

Classical boxplot classifies as potential outliers all points outside this interval (fence):

$$[Q_1 - 1.5 \text{ IQR}; Q_3 + 1.5 \text{ IQR}], \tag{6}$$

where Q_1 is the first quartile, Q_3 is the third quartile and $\text{IQR} = Q_3 - Q_1$ is the interquartile range. The adjusted boxplot uses this alternative fence:

$$[Q_1 - 1.5\, e^{a\text{MC}} \text{ IQR}; Q_3 + 1.5\, e^{b\text{MC}} \text{ IQR}], \tag{7}$$

where MC is the medcouple. For a univariate sample $\{x_1, \dots, x_n\}$, the medcouple is defined as $\text{MC} = \underset{x_i \leq Q_2 \leq x_j}{\mathit{median}}\, h(x_i, x_j)$, where Q_2 is the sample median and where for all $x_i \neq x_j$ the kernel function h is given by $h(x_i, x_j) = \frac{(x_j - Q_2) - (Q_2 - x_i)}{(x_j - x_i)}$.

When MC ≥ 0 (positive skewness) then $a = -4$ and $b = 3$ in Eq. 7. When MC < 0 (negative skewness) then $a = -3$ and $b = 4$ in Eq. 7 (D'Orazio, 2018).

When dealing with skewed distributions, the classical boxplot flags many regular observations as potential outliers. The adjusted boxplot modifies the whiskers such that the skewness is taken into account.

As we were developing the methodology, we carried out several preliminary tests comparing the performance of adjusted boxplots and tolerance intervals. From these experiments, we have verified that the adjusted boxplot gives a more accurate solution than the tolerance intervals, in terms of identifying more true outliers and fewer false outliers, so we will use the adjusted boxplot in our outlier detection method by default. Thus, the acronym for robust FADA aimed to outlier detection is RO-FADA.

2.5 Functional outlier detection methods

Several contributions devoted to detect outliers in functional data, together with their related software, can be found in the literature. We introduce them here because we will compare our methodology with all of them. The following is a detailed explanation of the methods. We give them a representative acronym (between parentheses). Most of them are based on the idea of functional depth (Hubert et al., 2017). Arribas and Romo (2014) took advantage of the relation between two functional depths and created a visualization, the outliergram (OUG), mainly focused on detecting shape outliers. OUG is in the R package **roahd** (Tarabelloni et al., 2018). Febrero et al. (2007) defined a new statistic following the idea of the likelihood ratio test (LRT). A somewhat similar idea was described in Febrero et al. (2008), where the functional depth for every curve is computed. If the depth is lower than a threshold, the curve is an outlier. The cutoff value is determined with a bootstrap procedure based on either trimming (TRIM) or weighting the sample (POND). LRT, TRIM and POND are in the R package **fda.usc** (Febrero-Bande and Oviedo de la Fuente, 2012). Sun and Genton (2011) presented the functional boxplot (FB), which is based on the center outward ordering induced by band depth for functional data. FB is in the R package **fda** (Ramsay et al., 2017). For multivariate functional data, the functional outlier map (FOM) was proposed by Hubert et al. (2015) as a graphical tool to detect outliers based on functional depths. FOM is in the R package **mrfDepth** (Segaert et al., 2017). It is also worth pointing out that the techniques contained in **roahd** are also implemented for multivariate functional data.

Other methods are based on other ideas. Rousseeuw and Leroy (1987) used the robust Mahalanobis distance but considering the functions as multivariate observations (RMAH). In Hyndman and Shahid Ullah (2007), a procedure using the robust principal components and the integrated squared forecast error was proposed (ISFE). RMAH and ISFE are in the R package **rainbow** (Shang and Hyndman, 2016) (LRT, TRIM and POND can also be used with this package). The methods that were proposed by Hyndman (2010) are like the functional highest density region boxplot (HDR). The R function to use HDR is extracted from **rainbow** (it cannot be directly used with the package). In Azcorra et al. (2018), the massive unsupervised outlier detection (MUOD) method was described. In this case, multivariate data are represented by means of parallel coordinates. Then, tools from FDA are used. MUOD is in the R package **muod**. Ramaswamy et al. (2000) proposed the k-nearest neighbors outlier detection method (kNNo). Each point's anomaly score is the distance to its kth nearest neighbor in the data set. Then, all points are ranked based on this distance. The higher an example's score is, the more anomalous it is. kNNo is in the R package **adamethods**. Millán-Roures et al. (2018) was a first attempt at using an ADA-based procedure for outlier detection in functional settings (FOADA). The detection is based on iteratively applying ADA and the robust Mahalanobis distance to the α coefficients. The algorithm iterates until no more outliers are found. Even though FOADA is the most similar approach to our new proposal, note the differences with our method, which is a non-iterative method, based on the use of the residuals (not on the α coefficients) and the adjusted boxplot.

3 Method: CRO-FADALARA (Cleaning and Robust Outlier FADA for LARge Applications)

Our approach consists in two phases. The first one is a cleaning procedure. The second one is RO-FADALARA (Robust Outlier FADA for LARge Applications). The outcome of the method is a set of outliers, together with the importance that each variable had in the outlier detection. As usual, archetypoids are also returned. Details of both phases are given next.

3.1 Phase I: Cleaning

The idea of this first phase is to clean the most evident outliers. Points that deviate either over the whole curve (amplitude outliers) or in a small interval (isolated outliers) will be the ones that this phase will mostly detect. This will allow RO-FADALARA to focus primarily on hidden behaviours (mainly shape outliers). The classical boxplot is used for this purpose. The curve is considered discretely and if the majority of the points are pointwise mild outliers, i.e., they lie between one point five times and three times the interquartile range below the lower quartile or above the upper quartile, then the curve is identified as an outlier function. We have considered the majority as more than 80% of the points. However, if one of the points is an extreme outlier, i.e., it is more than three times the interquartile range from the lower or upper quartile (beyond the outer fences), then we also consider it as an outlier function. Note that this cleaning step is also added to RO-FADA, which then becomes the CRO-FADA algorithm.

3.2 Phase II: RO-FADALARA

The CLARA (Clustering LARge Applications) algorithm (Kaufman and Rousseeuw, 1990) has inspired us to develop its counterpart method for archetypoids. The basic idea is to slice the data into several small samples instead of working with the entire data set. This is an iterative algorithm and the procedure in each iteration can be summarized as follows:

1. Obtain a sample s of the data set.
2. Apply R-FADA to s and obtain its archetypoids g_s.
3. For the whole data set, compute the matrix of coefficients α_{g_s}, the vector of residuals resid_{g_s} and the number RSS_{g_s}, with respect to g_s.
4. If RSS_{g_s} is smaller than the previously saved RSS, then save RSS_{g_s}, resid_{g_s} and g_s.

Once all the iterations are done, the outliers are computed using the adjusted boxplot with the norm of final resid_{g_s}. Algorithm 1 describes this procedure.

Note that steps 1 from 4 can serve not only for scaling R-FADA, but also for obtaining the archetypoids in a big dataset of any kind, by simply replacing R-FADA with R-ADA or any other ADA algorithm.

3.3 Computational details of CRO-FADALARA

The use of the sampling strategy enables archetypoid analysis to deal with large data and generate results in reasonable time. However, more time can still be saved by taking advantage of the ever-increasing processing power of modern computers. Nowadays, most computers are equipped with multiple processors (cores) with adequate amount of memory available. Parallel programming techniques benefit from multiple cores. Thus, parallelizing an algorithm can be used for returning quick outputs and make it efficient at the same time. According to these premises, we have also developed a parallel version of CRO-FADALARA. To that end, we have used the R package **doParallel**, which provides a stable framework to perform tasks in parallel by providing the ability to allocate cores to R (Ooi et al., 2017). Both non-parallel and parallel versions of CRO-FADALARA are available in the R package **adamethods**.

In practical terms, we would like to point out that the non-parallel and parallel algorithms can provide slightly different results. From our investigation, this is due to the different way in which the seed to ensure reproducible results is set in non-parallel and parallel scenarios. Since both algorithms show a good and very similar performance, we let the final decision about which algorithm to run be up to the user, only maybe at the expense of a somewhat less optimal solution.

Algorithm 1 *RO-FADALARA*

```
Input: data frame $d \in \mathbb{R}^{n \times p}$, number of archetypoids $g$, number of samples $N$, number
of random observations in each sample $s_{i,i=1,\ldots,N}$, $|s_i| = m$ $(m < n, N > 1)$
Note to the input: $m$ and $N$ are related via $N = 1 + (n - m)/(m - g)$.
Output: $g$ archetypoids and the set of outliers associated.
Ancillary function:

    $\underset{i=1,\ldots,N}{\text{r-fada}(s_i)} = \begin{cases} (i) \text{ Apply R-FADA on } s_i \text{ to compute } g_{s_i} \text{ archetypoids.} \\ (ii) \text{ For every observation in } d\text{, compute } \alpha_{g_{s_i}}, \text{resid}_{g_{s_i}} \text{ and } \text{RSS}_{g_{s_i}}. \end{cases}$

for $i = 1 \to N$ do
  if $i == 1$ then
    1. Generate $s_1 \subset d$, $|s_1| = r_1$, $r_1 = \{r_{11}, \ldots, r_{1m}\}$.
    2. Use r-fada($s_1$) to get $g_{s_1}$, $\alpha_{g_{s_1}}$, $\text{resid}_{g_{s_1}}$ and $\text{RSS}_{g_{s_1}}$.
  end if
  if $i == 2$ then
    3. Generate $s_2 \subset d$, $g_{s_1} \in s_2$, $|s_2| = g + r_2$, $r_2 = \{r_{21}, \ldots, r_{2(m-g)}\} \not\subseteq r_1$.
    4. Use r-fada($s_2$) to get $g_{s_2}$, $\alpha_{g_{s_2}}$, $\text{resid}_{g_{s_2}}$ and $\text{RSS}_{g_{s_2}}$.
  end if
  5. Retain $g_{aux} = g_{s_j, j=1 \text{ or } j=2}$ such that $\text{RSS}_{aux} = \min(\text{RSS}_{g_{s_1}}, \text{RSS}_{g_{s_2}})$.
  if $i > 2$ then
    6. Generate $s_i \subset d$, $g_{aux} \in s_i$, $|s_i| = g + r_i$,
       $r_i = \{r_{i1}, \ldots, r_{i(m-g)}\} \not\subseteq \{r_1, r_2, \ldots, r_{i-1}\}$.
    7. Use r-fada($s_i$) to get $g_{s_i}$, $\alpha_{g_{s_i}}$, $\text{resid}_{g_{s_i}}$ and $\text{RSS}_{g_{s_i}}$.
    if $\text{RSS}_{g_{s_i}} < \text{RSS}_{aux}$ then
      8. $\text{RSS}_{aux} = \text{RSS}_{g_{s_i}}$, $\text{resid}_{g_{aux}} = \text{resid}_{g_{s_i}}$ and $g_{aux} = g_{s_i}$.
    end if
  end if
end for
9. Apply the adjusted boxplot to the norm of $\text{resid}_{g_{aux}}$ to get $\text{outliers}_{g_{aux}}$.
return $g_{aux}$ and $\text{outliers}_{g_{aux}}$.
```

3.4 Variable importance

In the case of multivariate functional data, the importance that each variable had in the outlier detection process is also provided as complementary information. Both local and marginal relative importances are detailed. The local (casewise) relative importance refers to the outlier observation itself, the other observations are not considered. In computational terms, this means dividing the residual of the observation in each variable with respect to its total residual. In the marginal relative importance the other points are considered, since the value of the outlier observation is compared with the remaining points. In computational terms, this means dividing the residual of the observation with respect to the sum of all the residuals inside each variable. The important fact to remark here is that this procedure works because the functional variables are in the same scale, after standardizing. Otherwise, this interpretation could not be done.

3.5 Web applications

The web applications that we have developed allow the users to plot all the outliers and compare them with the non-outliers and the representative of each group of outliers. In any real situation, the number of anomalies is unknown, so there is no guarantee that the outlier detection method is providing the most accurate results. Furthermore, in anomaly detection settings, it is well-known that automatic methods should only be the first step, to be followed by human examination. In this context, the apps also offer to the domain experts the possibility of changing the category of the outliers curves in case some of them are not outliers, according to their expertise. A new database is automatically generated with the updated information. While interacting with the app, the users can also save any plot of interest and generate a document with them for further editing.

4 Results

To the best of our knowledge, there are not large time series databases publicly available, especially for the multivariate case. In spite of this, we have found two data sets with almost 14000 and 10000 observations, respectively, which we do believe that they can serve for illustration purposes. The first one comes from a gas sensor experiment where 8 variables were measured for six gaseous substances (Vergara et al., 2012; Rodríguez-Luján et al., 2014). This data set is particularly suitable to illustrate the noticeable reduction in computational time of CRO-FADALARA with respect to CRO-FADA. The second data set comes from the astronomy domain and is related to a common anomaly detection problem (Rebbapragada et al., 2009). In the field of astronomy, the development of powerful telescopes and detector technologies has led to a massive amount of data. One major challenge is to detect astronomical objects with anomalous physical properties. Therefore, this is an interesting case study to discuss the outliers that CRO-FADALARA discovers.

An electrocardiogram (ECG) is a measure of how the electrical activity of the heart changes over time. An ECG dataset is comprised of different components, or waves, that represent the electrical activity in specific regions of the heart. The processing of ECG records as functional data has become an important field of research. Hence, it is also worth using our methodology with this data, regardless of the size of the database.

In Section 4.1 we firstly carry out a numerical simulation with controlled data to evaluate the performance of CRO-FADA. All the scripts to reproduce these results can be found in the folder “Simulation” of the supplementary material code. Section 4.2 is devoted to the gas sensor data set (scripts in the folder “Drift_data”). Section 4.3 deals with the starlight-curve data set from the astrophysics domain (scripts in the folder “Starlight_data”). Section 4.4 shows the additional application to ECG data (scripts in the folder “ECG_data”). In short, we aim to answer the following questions:

– How does CRO-FADA perform compared to other functional data anomaly detection algorithms?
– How do the parallel and non-parallel CRO-FADALARA perform compared to CRO-FADA with large multivariate functional data sets?
– How can CRO-FADALARA be used in an anomaly detection problem?

The algorithms were executed in a workstation with an Intel i7 processor running at 2.40 GHz with 8 Gb of RAM and 3 cores under Linux (Fedora release 27) with R version 3.4.4. For both gas sensor and light-curves data sets, we have fixed the number of random observations in each sample to $m = 100$, the number of archetypoids to $g = 3$ and the quantile to compute the parameter c of the bisquare loss function to the 0.75 quantile (the third quartile).

4.1 Simulation study

In order to check the performance of our method, we have carried out a simulation study with several state-of-the-art benchmark methods. We have created a set of $F = 100$ functions observed at 50 equidistant points between 0 and 1. Initially, the 100 functions are generated from the main model $X_1(t) = 30t(1-t)^{3/2} + \epsilon(t)$. Then, we have replaced some of these functions by a number of shape, amplitude, isolated and shift outliers according to an outlier rate. We denote this rate as Δ. These contaminated functions are the outliers that the methods have to identify.

The shape outliers are generated using the same model as in Febrero et al. (2008), Fraiman and Svarc (2013), Arribas and Romo (2014) and Millán-Roures et al. (2018). It is defined as $X_2(t) = 30t^{3/2}(1-t) + \epsilon(t)$. Both in X_1 and X_2, $t \in [0,1]$ and $\epsilon(t)$ is a Gaussian process with zero mean and covariance function $\gamma(s,t) = 0.3exp(-|s-t|/0.3)$. The amplitude outliers are generated by adding 3 to the mean from the main model. The isolated outliers are generated by adding the values of a standard normal density to the first 14 observed points of the mean from the main model. The shift outliers are generated by translating the mean of the main model in -0.1 time units. Fig. 1 displays the type of outliers (obtained with /Simulation/do_plots.R). We have run 100 simulations. The accuracy of the methods is evaluated in terms of recall (percentage of true outliers detected), precision (percentage of true outliers in the set of outliers detected) and false positive rate (percentage of points that are not outliers, that are identified as outliers).

All the methods described in Section 2.5 are used as our benchmarking framework. The R functions related to the aforementioned packages have been used with their default parameters. Classical ADA with Frobenius norm was used. For kNNo, we have fixed an outlier rate of 0.01 for all cases (i.e., the believed proportion of outliers is 1% (1 of 100 functions)).

Table 1 shows the results for shape outliers, when the outlier rate is fixed to 0 (no outliers), 0.02 and 0.05. When there are no outliers, CRO-FADA shows a similar performance to the others. HDR and LRT do not find false

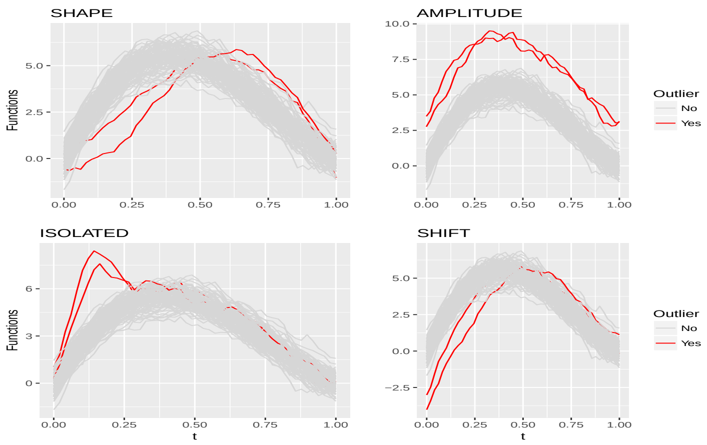

Fig. 1 Outlier types used in the simulation study. Outliers are in red color.

	Shape						
	$\Delta = 0$	$\Delta = 0.02$			$\Delta = 0.05$		
	FPR	TPR	Precision	FPR	TPR	Precision	FPR
CRO-FADA	2 (2.78)	98.5 (8.6)	76.8 (24.6)	1.08 (1.76)	94.4 (19.3)	87.3 (20.8)	0.88 (1.76)
OUG	2.6 (1.75)	97 (11.9)	53.9 (21.9)	2.27 (1.69)	96 (9)	76.3 (14.2)	1.81 (1.39)
RMAH	1.67 (1.36)	99.5 (5)	69.5 (22.7)	1.24 (1.14)	99.2 (3.9)	88.8 (11.5)	0.77 (0.84)
ISFE	3.91 (2.1)	64 (44.4)	27.9 (24)	3.83 (2.01)	48.8 (41.4)	34.9 (27.3)	4 (2.01)
LRT	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)
TRIM	0.94 (0.51)	92.5 (17.9)	99.2 (6)	0.02 (0.14)	63.2 (28.4)	99.2 (6)	0.02 (0.15)
POND	1.27 (0.98)	100 (0)	70.7 (20.2)	1.09 (0.92)	99.2 (3.9)	90.1 (10)	0.65 (0.68)
FOM	0.73 (1.11)	90.5 (21)	84.5 (20.4)	0.52 (0.73)	70.2 (30.7)	92.9 (12.3)	0.37 (0.66)
FB	0.01 (0.1)	31.5 (32.3)	98.2 (13.5)	0.01 (0.1)	29.8 (23)	99.6 (3.8)	0.01 (0.11)
HDR	0 (0)	79 (34.9)	79 (34.9)	0.43 (0.71)	74.8 (18.6)	74.8 (18.6)	1.33 (0.98)
MUOD	18.28 (3.7)	98.5 (8.6)	20.6 (10.1)	8.87 (3.16)	100 (0)	48.1 (11.5)	6.36 (2.95)
kNNo	1 (0)	50 (0)	100 (0)	0 (0)	20 (0)	100 (0)	0 (0)
FOADA	5.33 (3.16)	98.5 (11.1)	39.8 (20.7)	4.16 (2.85)	99 (5.2)	63.4 (16.7)	3.62 (2.59)

Table 1 Synthetic data with shape outliers for three different outlier rates (0%, 2% and 5%). Mean and standard deviation (in parentheses) of the True Positive Rate (TPR, also called Recall), Precision and False Positive Rate (FPR) over 100 simulation runs.

outliers. In fact, LRT does not find anything even when there are outliers. MUOD is obtaining a lot of false positives in all cases. This undermines its high recall when there are outliers in the data. When $\Delta = 0.02$, CRO-FADA has a very good recall, an acceptable precision and a small false positive rate. OUG, RMAH, TRIM, POND and FOM are the other methods that have a similar good performance in the three aspects. This CRO-FADA, OUG, RMAH and POND noteworthy performance remains when $\Delta = 0.05$. However, the recall of TRIM and FOM deteriorates. OUG has increased its precision in this case. FOADA shows a good recall when there are outliers in the data but its precision is not that good and its false positive rate is one of the highest in all cases.

Table 2 shows the results for amplitude, isolated and shift outliers, where the outlier rate is fixed to 0.05. For the amplitude outliers, CRO-FADA has a great performance in terms of its recall, as is the case with RMAH, POND, FOM, MUOD and FOADA. CRO-FADA has a smaller precision than RMAH,

	$\Delta = 0.05$								
	Amplitude			Isolated			Shift		
	TPR	Precision	FPR	TPR	Precision	FPR	TPR	Precision	FPR
CRO-FADA	100 (0)	77.0 (20.3)	2.22 (2.62)	92.2 (19.7)	90.5 (13.6)	0.71 (1.17)	94.2 (18)	89.6 (16.8)	0.68 (1.32)
OUG	0 (0)	0 (0)	2.66 (1.87)	21.4 (18.9)	36.5 (28.8)	1.96 (1.51)	86.2 (15)	74 (15.5)	1.82 (1.35)
RMAH	100 (0)	87.1 (11.7)	0.89 (0.89)	76.2 (28.7)	85.6 (18.1)	0.74 (0.9)	98.4 (5.5)	88.7 (12.5)	0.79 (0.95)
ISFE	38.6 (36.9)	31.4 (26.4)	3.91 (2)	99.4 (6)	61.3 (14.8)	3.77 (2.12)	50.2 (40.9)	34.8 (26.8)	4.28 (2.22)
LRT	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)
TRIM	82.6 (34.5)	98.4 (11.5)	0.02 (0.15)	67.2 (27.6)	98.3 (11.2)	0.03 (0.18)	55.6 (27.2)	98.5 (11.1)	0.02 (0.15)
POND	100 (0)	90.6 (10.4)	0.63 (0.76)	98.8 (4.8)	92.4 (9.8)	0.49 (0.68)	98.6 (5.1)	90.4 (10.2)	0.62 (0.69)
FOM	100 (0)	95.2 (9.1)	0.33 (0.66)	78.2 (30.6)	93.3 (12.3)	0.39 (0.79)	76 (28.1)	94.3 (9.9)	0.32 (0.57)
FB	94.6 (11)	99.8 (1.7)	0.01 (0.11)	55.6 (43.4)	98.6 (12)	0.01 (0.11)	33.6 (26.5)	99.6 (3.7)	0.01 (0.11)
HDR	65.4 (21.1)	65.4 (21.1)	1.82 (1.11)	56.2 (22.3)	56.2 (22.3)	2.31 (1.18)	74 (19)	74 (19)	1.37 (1)
MUOD	100 (0)	30 (5.6)	12.86 (3.34)	99.8 (2)	31.1 (7.8)	12.53 (3.84)	99 (4.4)	29.4 (8.2)	13.48 (3.89)
kNNo	20 (0)	100 (0)	0 (0)	19.8 (2)	99 (10)	0.01 (0.11)	19.8 (2)	99 (10)	0.01 (0.11)
FOADA	100 (0)	74.6 (15.9)	2.14 (1.68)	98.8 (6.9)	69.1 (16.6)	2.78 (2.06)	99.4 (4.5)	68.2 (17.2)	2.99 (2.27)

Table 2 Synthetic data with amplitude, isolated, and shift outliers, with an outlier rate of 5%. Mean and standard deviation (in parentheses) of the True Positive Rate (TPR, also called Recall), Precision and False Positive Rate (FPR) over 100 simulation runs.

POND and FOM. MUOD identifies all the true positives at the expense of identifying a lot of false positives (this is a constant behaviour, as can be seen in all the scenarios). For isolated outliers, CRO-FADA is also a liable method and only POND and FOADA remain as a valid competitors (although FOADA has a lower precision and a higher false positive rate). ISFE could be also highlighted, but its false positive rate is remarkable. For shift outliers, CRO-FADA and POND are again very good. FOADA is also good at recall, but not in terms of precision and false positives. RMAH is once again a powerful alternative. OUG performs here relatively well. It is important to remember that OUG was defined to detect shape outliers. Results of Table 1 and Table 2 can be obtained with /Simulation/do_simul_comp_tpr_pre_fpr.R.

Table 3 shows the results for all the outlier types together, where the outlier rate is fixed to 0.02. Real problems usually present outliers of different types. CRO-FADA and POND are again the best methods, followed by RMAH, HDR and FOM. MUOD and ISFE return too many false positives. It is worth mentioning that kNNo does not have a good performance when the real amount of outliers is not known. Results of Table 3 can be obtained with /Simulation/do_simul_comp_all_tpr_pre_fpr.R.

Overall, we can conclude that CRO-FADA is a very competitive method, in line with the methods that show the best performances in every case. In practical terms, we have checked that POND (and also TRIM) cannot deal with big data files and is computationally expensive. This is not the case for CRO-FADA when extended to CRO-FADALARA. Therefore, CRO-FADA becomes even more useful in a context of real data.

4.2 Gas sensor data

The first data set comes from the open UC Irvine Machine Learning Repository (Dua and Karra-Taniskidou, 2017). It contains 13910 measurements from 16 chemical sensors exposed to six gaseous substances at different concentration

	Shape, amplitude, isolated and shift		
	$\Delta = 0.02$		
	TPR	Precision	FPR
CRO-FADA	94.5 (11.6)	95.6 (8.9)	0.51 (1.32)
OUG	50 (10.7)	76 (16.6)	1.63 (1.37)
RMAH	89.5 (10.6)	93.2 (7.7)	0.61 (0.71)
ISFE	69.5 (16.2)	63.4 (14.6)	3.83 (2.06)
LRT	0 (0)	0 (0)	0 (0)
TRIM	34.1 (9.9)	99.7 (3.3)	0.01 (0.11)
POND	98.6 (4.3)	95.3 (6.3)	0.47 (0.64)
FOM	80.5 (18.3)	98.4 (4.8)	0.14 (0.43)
FB	55.4 (16.8)	99.9 (1.4)	0.01 (0.11)
HDR	84.8 (10.6)	84.8 (10.6)	1.33 (0.92)
MUOD	99.1 (3.2)	63.5 (9.2)	5.26 (2.13)
kNNo	12.5 (0)	100 (0)	0 (0)
FOADA	77.9 (10.3)	81.7 (12.7)	1.72 (1.43)

Table 3 Synthetic data with shape, amplitude, isolated, and shift outliers all together, with an outlier rate of 2%. Mean and standard deviation (in parentheses) of the True Positive Rate (TPR, also called Recall), Precision and False Positive Rate (FPR) over 100 simulation runs.

levels, producing a 16-channel time series [1](Vergara et al., 2012; Rodríguez-Luján et al., 2014). The six gases are Ammonia, Acetaldehyde, Acetone, Ethylene, Ethanol and Toluene. Two different types of features were considered in the creation of the data set: (i) the so-called steady-state feature (DR), which is the maximal resistance change with respect to the baseline and its DR normalized version; (ii) an aggregate of variables reflecting the sensor dynamics of the increasing/decreasing transient portion of the sensor response during the measurement process. Values 0.1, 0.01 and 0.001 were set to obtain three different feature values both from the increasing and decreasing portion of the sensor response. Thus, a total of 8 features are extracted from each 16-sensor time series: DR_j, |DR|_j, EMAi0.001_j, EMAi0.01_j, EMAi0.1_j, EMAd0.001_j, EMAd0.01_j, EMAd0.1_j ($j = 1, \ldots, 16$). EMAi and EMAd refer to the increasing and decreasing situation, respectively. In practical terms, the input data for CRO-FADA and the parallel and non-parallel versions of CRO-FADALARA is an eight-dimensional array with 13910 rows and 16 columns. Since gas sensor curves are non-periodic time series we have used the splines basis. The number of bases is 10. Table 4 shows the RSS associated with the three algorithms. These results can be obtained with /Drift_data/do_drift_fada.R, /Drift_data/do_drift_fadalara.R and Drift_data/do_drift_fadalara_par.R. As expected, the smallest RSS is for FADA because the minimization is with respect to the whole data set, not with respect to smaller samples, but it took three days to be computed. CRO-FADALARA reduces computational time dramatically, only at the expense of a slightly less optimal solution. The RSS of the parallel version is also a bit less optimal than that of the non-parallel, but it was the fastest option. In order to cope with this level of inaccuracy, the users

[1] `http://archive.ics.uci.edu/ml/datasets/Gas+Sensor+Array+Drift+Dataset+at+Different+Concentrations`

	Comp.time	RSS
CRO-FADA	3 days	0.249
Non-parallel CRO-FADALARA	25 minutes	0.258
Parallel CRO-FADALARA	14 minutes	0.261

Table 4 RSS and time of CRO-FADA and non-parallel and parallel CRO-FADALARA.

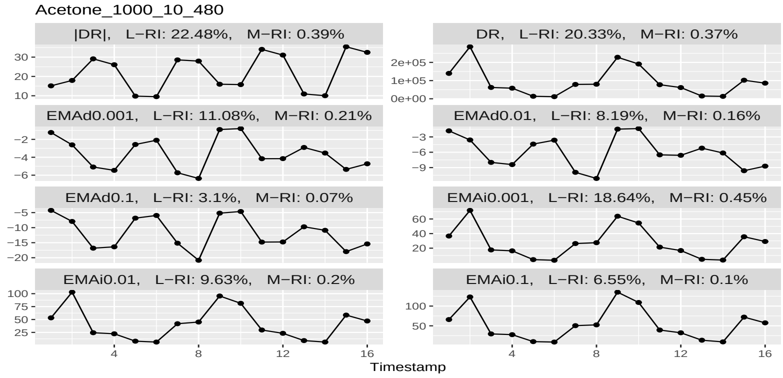

Fig. 2 Illustration of an outlier curve detected by CRO-FADALARA in the gas sensor data. The variable importance of each variable in the outlier detection procedure is indicated. L-RI means local relative importance and M-RI means marginal relative importance.

can change any mistakenly flagged outliers in the web application. The final point we would like to emphasize here is that the more cores the computer has, the faster the parallel CRO-FADALARA will return results.

In the interests of illustration, Fig. 2 displays the activity of one of the outliers detected by non-parallel CRO-FADALARA in every variable. The variable importance in the outlier detection procedure is also indicated. For this observation, DR, |DR| and EMAi0.001 were the most important variables. To make the outlier inspection more effective, Fig. 3 displays the DR and |DR| values for the same outlier, together with a given non-outlier and the archetypoid most related to the outlier (according to the α values of the outlier. As a reminder, the α coefficients represent how much each archetypoid contributes to the approximation of each observation.). The outliers can be grouped according to their largest α with respect the archetypoids. This plot makes it easy to compare the performance of every curve in every single variable. For this particular example, we see that the outlier is indeed having a different performance in both steady-state features with respect to the non-outlier. Only two variables were displayed to shorten the explanation [2]. Results of Fig. 2 and Fig. 3 can be obtained with /Drift_data/do_drift_fadalara_plots.R.

[2] Run in R these two commands for inspecting all the results:
library(shiny) ; runUrl('path to/Drift_data_app.zip')

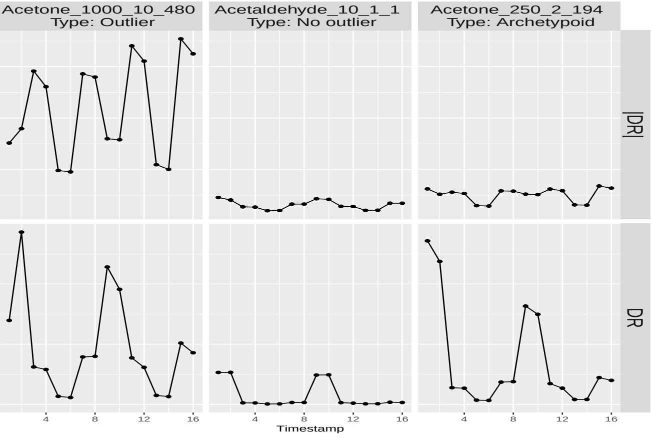

Fig. 3 Plot for an easy comparison of outliers, non-outliers and the corresponding representative of the outliers group. Illustration for DR and |DR|.

4.2.1 Comparison of the cleaning step with other outlier detection methods

Following a Referee's suggestion, we have compared the performance of the cleaning step of CRO-FADALARA with two graphical methods included in the R package **roahd**, namely, the functional boxplot and the multivariate outliergram. As explained in the vignette of **roahd**, both methods are useful for robustifying a functional dataset by removing amplitude (with the functional boxplot) and shape (with the outliergram) outliers (Tarabelloni et al., 2018). In terms of the software used, version 1.4 of **roahd** has been installed from source and used for this analysis.

The functional boxplot is obtained by ranking functions from the center of the distribution outwards. The procedure includes these steps: (i) computing depth values; (ii) computing the region of 50% most central functions, and (iii) inflating such region by a user given factor. Any function beyond these boundaries is identified as an outlier. The multivariate outliergram extends the univariate outliergram to the case of multivariate functional datasets. In the univariate case, the outliergram is based on the computation of the Modified Band Depth and Modified Epigraph Index. Such pairs are compared to a limiting parabola, where they should be located in case of non-crossing data. Outliers are then indicated by using a thresholding rule.

Default parameters have been used when calling these two methods with **roahd**. The functional boxplot returns 815 outliers and the multivariate outliergram, 237. As a whole, both methods identify a total of 894 unique outliers. The cleaning step of our method detects 723. From these numbers, we get that a total of 432 outliers are common to both approaches. As a conclusion, the union of the functional boxplot and multivariate outliergram seems to be less conservative that our approach, with the risk of detecting too many false positives. This experiment and the analysis discussed in Section 4.1 allow us to compare several methods. All the results obtained encourage us to say that we have developed a reliable method. These results can be obtained with /Drift_data/do_drift_roahd.R.

4.3 Light-curve data

This second data set comes from the open UEA & UCR Time Series Classification Repository (Bagnall et al., 2018). It contains 9236 starlight-curves of length 1024, where 1329 are Cepheid (CEPH), 2580 are Eclipsing Binary (EB) and 5327 are RR Lyrae (RRL) (Rebbapragada et al., 2009, Section 6.2). This is a univariate time series data. CEPH, EB and RRL are common types of periodic variable stars and their analysis in terms of anomaly detection is very important to astronomy. CEPH are radially pulsating supergiant stars, EB are binary star systems in which the orbit plane of two stars lies very close. RRL are radially pulsating stars. A light-curve is a real-valued series measuring the magnitude of light in each image captured of the night sky over time, together with its observational error.

In this case, the input for CRO-FADALARA is a univariate data matrix with 9236 and 1024 columns. Since light-curves are periodic time series we have used the Fourier basis. The number of bases is 15. We have applied both non-parallel and parallel CRO-FADALARA. Non-parallel CRO-FADALARA has returned an RSS of 14.3 in 6 minutes. The parallel version, an RSS of 14.9 in 3.5 minutes. These results can be obtained with /Starlight_data/do_starlight_fadalara.R and /Starlight_data/do_starlight_fadalara_par.R. We discuss the non-parallel results, since it returned the most optimal solution. 8 CEPH, 109 EB and 25 RRL have been identified as outliers. Fig. 4 shows a typical light-curve from each star class, together with an example of the type of curves identified as outliers (obtained with /Starlight_data/do_starlight_fadalara_plots.R). CEPH and RRL have similar shapes because they are both radially pulsating stars. On the contrary, the outliers identified for them do not follow the same periodic pattern. Regarding EB, most of the EB curves show two peaks, one of them always around the time point 250. In general, the type of EB outliers show a more noisy behaviour. However, for EB curves the difference between outliers and non-outliers is not as clear as happens with CEPH and RRL [3] .

4.4 ECG data

The dataset that we have used is contained in the R package **roahd** and collects the 8-lead ECG traces of 50 healthy subjects. As explained in **roahd**, the 8 leads are, in order, V1, V2, V3, V4, V5, V6, D1 and D2. The signals have been registered and smoothed over an evenly spaced grid of 1024 time

[3] Run in R these two commands for inspecting all the results:
library(shiny) ; runUrl('path to/Starlight_data_app.zip')

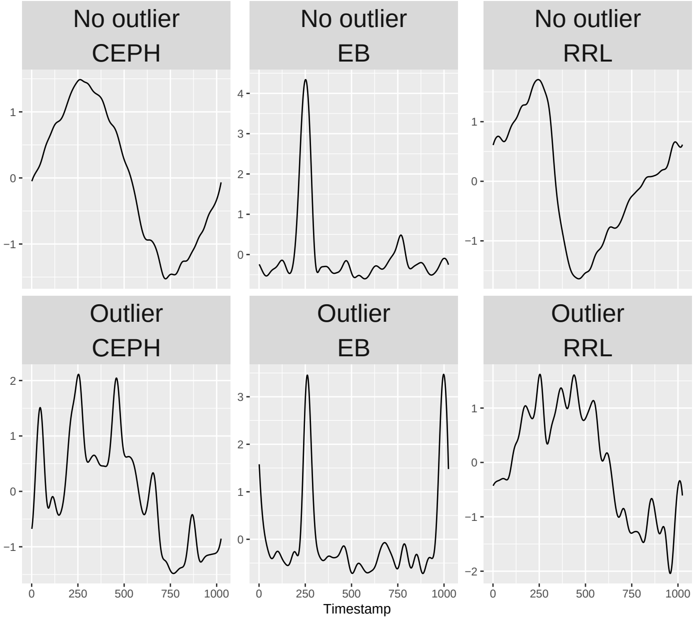

Fig. 4 Illustration of the shape of the outliers and non-outliers for the three types of stars, Cepheid (CEPH), Eclipsing Binary (EB) and RR Lyrae (RRL).

points at 1kHz. Due to the small size of this dataset, we have used the CRO-FADA algorithm. We have assumed that these ECG curves are non-periodic time series, so we have used the splines basis. The number of bases is 15. Four outliers have been identified. In the interests of illustration, Fig. 5 displays the ECG signals of one of the outliers in every lead. For this observation, leads V3 and V4 were the most important variables, as can be clearly appreciated in their plot facets. Interestingly, the heart activity of this subject in D1 (the third most important variable) is not a smooth curve, but rather shows some rough peaks. Our method also appears to be able to detect these patterns. These results can be reproduced with the scripts of the folder ECG_data.

5 Conclusions

This paper has addressed two open questions in archetypoid analysis: functional anomaly detection and scalability. Robust functional archetypoids have been used in combination with the adjusted boxplot for detecting outliers in highly skewed residual distributions. This new method is CRO-FADA. Furthermore, we have presented a new algorithm based on a sampling strategy, aimed at obtaining archetypoids and outliers from large databases in a faster and more reasonable period of time. This new algorithm is CRO-FADALARA. A parallel version of CRO-FADALARA has also been developed aimed to fur-

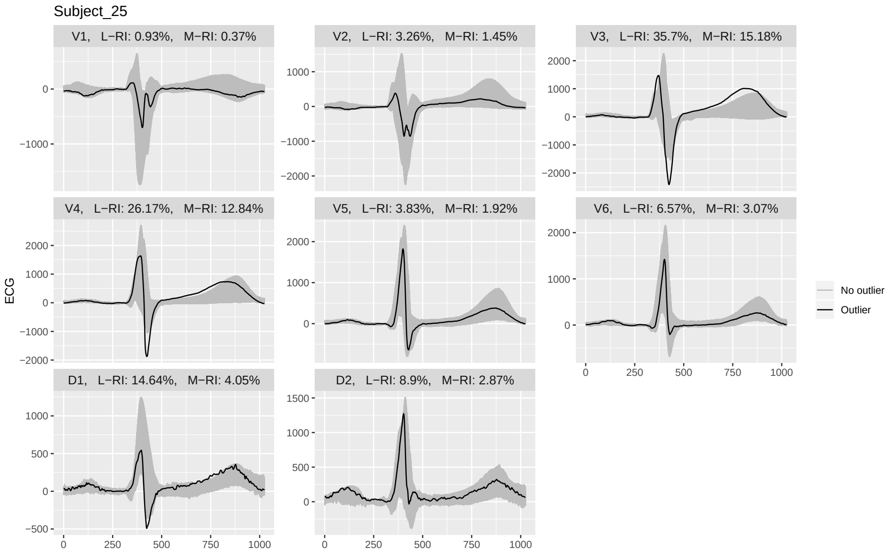

Fig. 5 One of the outliers identified in the ECG data. The outlier is in black color and the non-outlier curves are in grey. For this observation, leads V3, V4 and D1 were the most important variables.

ther decrease the computational time, by taking advantage of the multiple cores of a modern computer.

In many practical problems, individual observations are functions of time, observed at a set of discrete time points. Each curve represents the activity of a certain process of interest for each individual. If the process is known to be continuous and smooth, curves can be treated as functional data. This study has been focused on the FDA scenario. Outlier detection is an important topic also in FDA. We have used synthetic data to compare CRO-FADA with all the methods already developed for outlier detection in functional data. Different simulations have been executed for the different types of functional outliers, with several outlier rates. Our proposal shows one of the best performances in all cases, in terms of recall, precision and false positive rate.

The lack of publicly available large multivariate time series databases is a problem to check the performance of new algorithms. We have applied CRO-FADALARA to two open databases with almost 140000 and 10000 observations, respectively. The first data set is produced from an experiment with six gases where 8 features were extracted from a 16-channel time series. We have used this data set to compare the results obtained with CRO-FADA and CRO-FADALARA in terms of computational time and residual sum of squares. CRO-FADALARA dramatically reduces time, at the expense of a slightly less optimal results. The second application of CRO-FADALARA has concerned a data set containing measurements from starlight-curves, related to a common anomaly detection problem in the field of astronomy, where detecting anomalous patterns is very important. As demonstrated, CRO-FADALARA can be very useful in anomaly detection problems. CRO-FADALARA can be used both in the univariate and multivariate case, which highlights its added value. A third application with a small sample of ECG data, which is a classical field of research in functional data applications, has been also discussed. Thanks to this study, it is possible to say that archetypoid analysis has become a fully versatile method, which can be used with either traditional multivariate or functional data (both univariate and multivariate), either small or large, and with different objectives, namely the identification of extreme observations or outlier detection. All the data and R code, including the new R package **adamethods**, are freely available. We aim to incorporate new strategies in CRO-FADALARA to get results as fast and optimal as possible. Interesting future work is to delve into how to reduce the computational burden of archetypes, where there have been some initial attempts (Mair et al., 2017; Chen et al., 2014).

Acknowledgements GV worked on the first version of the manuscript as a postdoctoral scholarship holderin international mobility at KU Leuven and acknowledges support from SBO grant HYMOP (150033) of the Research Foundation-Flanders (FWO-Vlaanderen). GV thanks: (i) Wannes Meert and Jesse Davis for the follow-up in the context of the HYMOP project and the suggestion of computing the variable importance; (ii) Jordi Fonollosa for the help with the gas sensor data; (iii) Sebastian Mair for the frame-based data factorization code. IE was supported by DPI2017-87333-R from the Spanish Ministry of Science, Innovation and Universities (AEI/FEDER, EU) and UJI-B2017-13 from Universitat Jaume I. The authors also thank the anonymous reviewers for their comments, and the UCI Machine Learning and UEA & UCRTime Series Classification repositories for providing open data.

References

Alcacer, A., Epifanio, I., Ibáñez, M., Simó, A., Ballester, A.: A data-driven classification of 3D foot types by archetypal shapes based on landmarks. PLOS ONE **15**(1), e0228016 (2020). https://doi.org/10.1371/journal.pone.0228016

Arribas-Gil, A., Romo, J.: Shape outlier detection and visualization for functional data: the outliergram. Biostatistics **15**(4), 603–619 (2014). https://doi.org/10.1093/biostatistics/kxu006

Azcorra, A., Chiroque, L., Cuevas, R., Fernández Anta, A., Laniado, H., Lillo, R., Romo, J., Sguera, C.: Unsupervised Scalable Statistical Method for Identifying Influential Users in Online Social Networks. Scientific Reports **8**, 1–7 (2018). https://doi.org/10.1038/s41598-018-24874-2, https://github.com/luisfo/muod.outliers

Bagnall, A., Lines, J., Vickers, W., Keogh, E.: The UEA & UCR Time Series Classification Repository. www.timeseriesclassification.com (2018)

Beaton, A., Tukey, J.: The Fitting of Power Series, Meaning Polynomials, Illustrated on Band-Spectroscopic Data. Technometrics **16**(2), 147–185 (1974). https://doi.org/10.1080/00401706.1974.10489171

Cabero, I., Epifanio, I.: Archetypal analysis: an alternative to clustering for unsupervised texture segmentation. Image Analysis & Stereology **38**, 151–160 (2019). https://doi.org/10.5566/ias.2052

Cabero, I., Epifanio, I.: Finding archetypal patterns for binary questionnaires. SORT **44**(1), in press (2020). https://arxiv.org/abs/2003.00043

Chang, W., Cheng, J., J.J., A., Xie, Y., McPherson, J.: shiny: Web Application Framework for R (2017). URL https://CRAN.R-project.org/package=shiny. R package version 1.0.5

Chen, Y., Mairal, J., Harchaoui, Z.: Fast and Robust Archetypal Analysis for Representation Learning. In: CVPR 2014 - IEEE Conference on Computer Vision and Pattern Recognition, pp. 1478–1485 (2014). https://doi.org/10.1109/CVPR.2014.192

Cutler, A., Breiman, L.: Archetypal Analysis. Technometrics **36**(4), 338–347 (1994). https://doi.org/10.2307/1269949

D'Orazio, M.: univOutl: Detection of Univariate Outliers (2018). URL https://CRAN.R-project.org/package=univOutl. R package version 0.1-4

Dua, D., Karra-Taniskidou, E.: UCI machine learning repository. University of California, Irvine, School of Information and Computer Sciences (2017). http://archive.ics.uci.edu/ml

Epifanio, I.: Functional archetype and archetypoid analysis. Computational Statistics and Data Analysis **104**, 24–34 (2016). https://doi.org/10.1016/j.csda.2016.06.007

Epifanio, I., Ibáñez, M., Simó, A.: Archetypal shapes based on landmarks and extension to handle missing data. Advances in Data Analysis and Classification **12**, 705–735 (2018). https://doi.org/10.1007/s11634-017-0297-7

Epifanio, I., Ibáñez, M., Simó, A.: Archetypal analysis with missing data: see all samples by looking at a few based on extreme profiles. The American Statistician **72**, 169–183 (2020). https://doi.org/10.1080/00031305.2018.1545700

Eugster, M., Leisch, F.: Weighted and robust archetypal analysis. Computational Statistics and Data Analysis **55**, 1215–1225 (2011). https://doi.org/10.1016/j.csda.2010.10.017

Febrero, M., Galeano, P., González-Manteiga, W.: A functional analysis of NO_x levels: location and scale estimation and outlier detection. Computational Statistics **22**(3), 411–427 (2007). https://doi.org/10.1007/s00180-007-0048-x

Febrero, M., Galeano, P., González-Manteiga, W.: Outlier detection in functional data by depth measures, with application to identify abnormal NO_x levels. Environmetrics **19**, 331–345 (2008). https://doi.org/10.1002/env.878

Febrero-Bande, M., Oviedo de la Fuente, M.: Statistical Computing in Functional Data Analysis: The R Package fda.usc. Journal of Statistical Software **51**(4), 1–28 (2012). http://www.jstatsoft.org/v51/i04/

Fraiman, R., Svarc, M.: Resistant estimates for high dimensional and functional data based on random projections. Computational Statistics & Data Analysis **58**, 326–338 (2013). https://doi.org/10.1016/j.csda.2012.

09.006

Hamilton, J.: Time Series Analysis. Princeton Univ Press. (1994)

Hubert, M., Rousseeuw, P., Segaert, P.: Multivariate functional outlier detection. Statistical Methods & Applications **24**(2), 177–202 (2015). https://doi.org/10.1007/s10260-015-0297-8

Hubert, M., Rousseeuw, P., Segaert, P.: Multivariate and functional classification using depth and distance. Advances in Data Analysis and Classification **11**, 445–466 (2017). https://doi.org/10.1007/s11634-016-0269-3

Hyndman, R., Shahid Ullah, M.: Robust forecasting of mortality and fertility rates: A functional data approach. Computational Statistics & Data Analysis **51**(10), 4942–4956 (2007). https://doi.org/10.1016/j.csda.2006.07.028

Hubert, M., Vandervieren, E.: An adjusted boxplot for skewed distributions. Computational Statistics and Data Analysis **52**, 5186–5201 (2008). https://doi.org/10.1016/j.csda.2007.11.008

Hyndman, R.: Rainbow Plots, Bagplots, and Boxplots for Functional Data. Journal of Computational and Graphical Statistics **19**(1), 29–45 (2010). https://doi.org/10.1198/jcgs.2009.08158

Kaufman, L., Rousseeuw, P.: Finding Groups in Data. An Introduction to Cluster Analysis. John Wiley & Sons, Inc. (1990)

Mair, S., Boubekki, A., Brefeld, U.: Frame-based data factorizations. In: Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, pp. 2305–2313 (2017). http://proceedings.mlr.press/v70/mair17a/mair17a.pdf

Millán-Roures, L., Epifanio, I., Martínez, V.: Detection of Anomalies in Water Networks by Functional Data Analysis. Mathematical Problems in Engineering **2018**, 1–14 (2018). https://doi.org/10.1155/2018/5129735

Moliner, J., Epifanio, I.: Robust multivariate and functional archetypal analysis with application to financial time series analysis. Physica A: Statistical Mechanics and its Applications **519**, 195–208 (2019). https://doi.org/10.1016/j.physa.2018.12.036

Nun, I., Pichara, K., Protopapas, P., Kim, D.W.: Supervised detection of anomalous light curves in massive astronomical catalogs. The Astrophysical Journal **793**(23), 1–16 (2014). http://stacks.iop.org/0004-637X/793/i=1/a=23

Ooi, H., Microsoft Corporation, Weston, S., Tenenbaum, D.: doParallel: Foreach Parallel Adaptor for the 'parallel' Package (2017). URL https://CRAN.R-project.org/package=doParallel. R package version 1.0.11

R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2018). URL https://www.R-project.org/

Ramaswamy, S., Rastogi, R., Shim, K.: Efficient algorithms for mining outliers from large data sets. In: SIGMOD '00 Proceedings of the 2000 ACM SIGMOD international conference on Management of data, pp. 427–438 (2000). https://doi.org/10.1145/342009.335437

Ramsay, J.O., Silverman, B.: Functional Data Analysis. 2nd Edition, Springer (2005)

Ramsay, J.O., Hooker, G., Graves, S.: Functional Data Analysis with R and MATLAB. Springer (2009)

Ramsay, J.O., Wickham, H., Graves, S., Hooker, G.: fda: Functional Data Analysis (2017). R package version 2.4.7, `https://CRAN.R-project.org/package=fda`

Rebbapragada, U., Protopapas, P., Brodley, C., Alcock, C.: Finding anomalous periodic time series. An application to catalogs of periodic variable stars. Machine Learning (2009). `https://doi.org/10.1007/s10994-008-5093-3`

Rodríguez-Luján, I., Fonollosa, J., Vergara, A., Homer, M., Huerta, R.: On the calibration of sensor arrays for pattern recognition using the minimal number of experiments. Chemometrics and Intelligent Laboratory Systems **130**, 123–134 (2014). `https://doi.org/10.1016/j.chemolab.2013.10.012`

Rousseeuw, P., Leroy, A.: Robust Regression and Outlier Detection. John Wiley & Sons, Inc. (1987)

Segaert, P., Hubert, M., Rousseeuw, P., Raymaekers, J.: mrfDepth: Depth Measures in Multivariate, Regression and Functional Settings (2017). R package version 1.0.6, `https://CRAN.R-project.org/package=mrfDepth`

Shang, H.L., Hyndman, R.J.: rainbow: Rainbow Plots, Bagplots and Boxplots for Functional Data (2016). R package version 3.4, `https://CRAN.R-project.org/package=rainbow`

Sinova, B., González Rodríguez, G., Van Aelst, S.: M-estimators of location for functional data. Bernouilli **24**(3), 2328–2357 (2018). `https://doi.org/10.3150/17-BEJ929`

Sun, Y., Genton, M.: Functional Boxplots. Journal of Computational and Graphical Statistics **20**(2), 316–334 (2011). `https://doi.org/10.1198/jcgs.2011.09224`

Sun, W., Yang, G., Wu, K., Li, W., Zhang, D.: Pure endmember extraction using robust kernel archetypoid analysis for hyperspectral imagery. ISPRS Journal of Photogrammetry and Remote Sensing **131**, 147–159 (2017). `https://doi.org/10.1016/j.isprsjprs.2017.08.001`

Tarabelloni, N., Arribas-Gil, A., Ieva, F., Paganoni, A.M., Romo, J.: roahd: Robust Analysis of High Dimensional Data (2018). R package version 1.4, `https://CRAN.R-project.org/package=roahd`

Thieler, A., Fried, R., Rathjens, J.: RobPer: An R Package to Calculate Periodograms for Light Curves Based on Robust Regression. Journal of Statistical Software **69**(9), 1–37 (2016). `https://doi.org/10.18637/jss.v069.i09`

Vergara, A., Vembu, S., Ayhan, T., Ryan, M., Homer, M., Huerta, R.: Chemical gas sensor drift compensation using classifier ensembles. Sensors and Actuators B: Chemical **166**, 320–329 (2012). `https://doi.org/10.1016/j.snb.2012.01.074`

Vinué, G., Epifanio, I., Alemany, S.: Archetypoids: A new approach to define representative archetypal data. Computational Statistics and Data Analysis **87**, 102–115 (2015). https://doi.org/10.1016/j.csda.2015.01.018

Vinué, G., Epifanio, I.: Archetypoid analysis for sports analytics. Data Mining and Knowledge Discovery **31**(6), 1643–1677 (2017). https://doi.org/10.1007/s10618-017-0514-1

Vinué, G.: Anthropometry: An R Package for Analysis of Anthropometric Data. Journal of Statistical Software **77**(6), 1–39 (2017). https://doi.org/10.18637/jss.v077.i06

Vinué, G., Epifanio, I.: Forecasting basketball players' performance using sparse functional data. Statistical Analysis and Data Mining: The ASA Data Sci Journal **12**(6), 534–547 (2019). https://doi.org/10.1002/sam.11436

Young, D.: tolerance: An R package for estimating tolerance intervals. Journal of Statistical Software **36**(5), 1–39 (2010). https://doi.org/10.18637/jss.v036.i05