# Topological Analysis for Detecting Anomalies (TADA) in dependent sequences: application to Time Series

Frédéric Chazal, Clément Levrard, Martin Royer

Journal of Machine Learning Research 25 (2024) 1-48 Submitted 6/24; Revised 9/24; Published 12/24

# Topological Analysis for Detecting Anomalies (TADA) in dependent sequences: application to Time Series.

**Frédéric Chazal** FREDERIC.CHAZAL@INRIA.FR
*Inria Saclay*
*91120, Palaiseau, France*

**Clément Levrard** CLEMENT.LEVRARD@UNIV-RENNES1.FR
*Université de Rennes*
*35000, Rennes, France*

**Martin Royer** MARTIN.ROYER@INRIA.FR
*Inria Saclay, IRT SystemX*
*91120, Palaiseau, France*

**Editor:** Sayan Mukherjee

## Abstract

This paper introduces a new methodology based on the field of Topological Data Analysis for detecting structural anomalies in dependent sequences of complex data. A motivating example is that of multivariate time series, for which our method allows to detect global changes in the dependence structure between channels. The proposed approach is lean enough to handle large scale data sets, and extensive numerical experiments back the intuition that it is more suitable for detecting global changes of correlation structures than existing methods. Some theoretical guarantees for quantization algorithms based on dependent sequences are also provided.

**Keywords:** Topological Data Analysis, Unsupervised Learning, Anomaly Detection, Multivariate Time Series, $\beta$-mixing coefficients.

## 1 Introduction

Monitoring the evolution of the global structure of time-dependent complex data, such as, e.g., multivariate time series or dynamic graphs, is a major task in real-world applications of machine learning. The present work considers the case where the global structure of interest may be encoded by a persistence diagram, with a particular attention to the weighted dynamic graph encoding the dependence structure between the different channels of a multivariate time series. Such a situation may be encountered in various fields, such as e.g. EEG signal analysis Mohammed et al. (2023) or monitoring of industrial processes Li et al. (2022), and has recently given rise to an abundant literature - see, e.g. Zheng et al. (2023); Ho et al. (2023) and references therein.

The specific monitoring task addressed in this paper is unsupervised anomaly detection, that is to detect when the global structure is far enough from a so-called 'normal' regime to be considered as anomalous. In the multivariate time series framework, this amounts to detect when the dependence patterns between channels depart from the normal regime. From the mathematical point of view, this problem, in its whole generality, is ill-posed: one has

access to unlabeled data, in which it is tacitly assumed that the normal regime is prominent, the goal is then to label data points as normal or abnormal in a fully unsupervised way. In this sense, anomaly detection shows clear connection with outlier detection in robust machine learning (for instance robust clustering as in Brécheteau et al. 2021; Jana et al. 2024). For more insights and benchmarks on the specific problem of anomaly detection in time series the reader is referred to Paparrizos et al. (2022b) (univariate case) and to Wenig et al. (2022) for the multivariate case.

We introduce a new framework, coming with mathematical guarantees, based on the use of Topological Data Analysis (TDA), a field that has know an increasing interest to study complex data - see, e.g. Chazal and Michel (2021) for a general introduction. Application of TDA to anomaly detection in time series have raised a recent and growing interest: in medicine (Dindin et al. 2019; G. et al. 2014; Chrétien et al. 2024), cyber security (Bruillard et al. 2016), to name a few. Some general surveys on TDA applications to time series may be found in Ravishanker and Chen (2019); Umeda et al. (2019).

In this paper, the proposed approach proceeds in three steps. First, the time-dependence structure of a time series is encoded as a dynamic graph in which each vertex represents a channel of the time series and each weighted edge encodes the dependence between the two corresponding vertices over a time window. Persistent homology, a central theory in TDA, is then used to robustly extract the global topological structure of the dynamic graph as a sequence of so-called persistence diagrams. Second, we introduce a specific encoding of persistence diagrams, that has been proven efficient and simple enough to face large-scale problems in the independent case (Chazal et al. 2021). Finally, we produce a topological anomaly score based on this encoding.

The two last steps may be applied to any dependent sequence of persistence diagrams, encompassing diagrams generated from a sequence of filtered simplicial complexes in general. Monitoring the dependence structure of multivariate time series being the practical motivation of this work, the details of the full process (from raw data to anomaly scores) are given in this setting.

### 1.1 Contributions

Our main contributions are the following.

- We produce a new machine learning methodology for learning the normal topological behavior of complex data. This methodology is unsupervised, it does not need to be calibrated on uncorrupted data, as long as the amount of corrupted data remains limited with respect to the uncorrupted one. The proposed pipeline is easy to implement, flexible and can be adapted to different specific applications and framework involving graph data or more general topological data;

- In the multivariate time series case, the captured information is empirically shown to be different and, in several cases, more informative than the one captured by other state-of-the-art approaches. This methodology is lean by design, and enjoys novel interpretable properties with regards to anomaly detection that have never appeared in the literature, up to our knowledge;

- The resulting method can be deployed on architectures with limited computational and memory resources: once the training phase has been completed, the anomaly detection procedure relies on a few memorized parameters and simple persistent homology computations. Moreover, this procedure does not require any storage of previously processed data, preventing privacy issues;

- Some convergence guarantees for quantization algorithms - used to vectorize topological information - in the dependent case are proven. These results do not restrict to the specific setting of the paper and may be generalized in the general framework of $M$-estimation with dependent observations;

- Extensive numerical investigation has been carried out in three different frameworks. First on new synthetic data directly inspired from brain modeling problems as exposed in Bourakna et al. (2022), that are particularly suited for TDA-based methods and may be used as novel benchmark. They are added to the public The GUDHI Project (2015) library at `github.com/GUDHI/gudhi-data`. Second on the comprehensive benchmark 'TimeEval', that encompasses a large array of synthetic data sets Schmidl et al. (2022). And third on a real-case 'Exathlon' data set from Jacob et al. (2021-07). All of these experiments assess the relevance of our approach compared with current state-of-the-art methods. Our procedure, originating from concrete industrial problems, is implemented and has been deployed within the Confiance.ai program and an open-source release is incoming. Its implementation involves only standard, tested machine learning tools.

### 1.2 Organization of the paper

A complete description of the proposed methodology is provided in Section 2. Details on the persistence diagrams construction are given for multivariate time series, providing a comprehensive pipeline to build an anomaly score from raw data in this case. Next, Section 3 theoretically grounds the centroid computation step as well as the anomaly test introduced at the end of the methodological section. Section 4 gathers the numerical experiments in the three different settings introduced above (synthetic TDA-friendly, TimeEval synthetic, real Exathlon data). Proofs of our results are postponed to Section 7.

## 2 Methodology

This section describes the process to build an anomaly score from a dependent sequence of topologically structured data. Our method can be roughly decomposed into three steps (depicted in Figure 1):

- *Step 1*: From raw data build a sequence of topological descriptors of the structure via persistent homology. The output of this step is a dependent sequence of persistence diagrams (see Section 2.1.2).

- *Step 2*: Convert the sequence of persistence diagrams into a vector-valued sequence, using the approach depicted in Chazal et al. (2021) (see Section 2.2.2).

- *Step 3*: Build from the vector-valued sequence an anomaly score function (based on robust Mahalanobis distance, see Section 2.3).

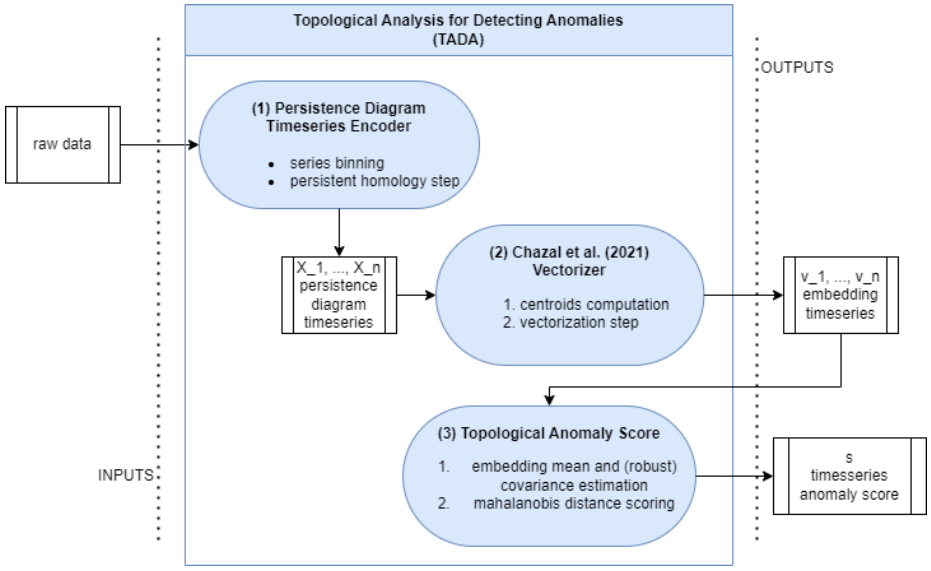


Figure 1: TADA general scheme for producing anomaly scores with topological information from the original time series.

To be consistent with the applications exposed in Section 4, details of these steps will be given in the case where original data consists of multivariate time series $(Y_t)_{t\in[0,L]} \in \mathbb{R}^D \times [0, L]$ and the anomalies to be detected pertain to the dependence structure between channels. This particular setting impacts Step 1 only. In other situations, Step 2 and 3 may apply as such once suitable persistence diagrams are built from raw data.

### 2.1 Step 1: encode the dependence structure via persistence diagrams

Our first step may be summarized via the following Algorithm 1:

**Algorithm 1:** Persistence diagrams computation from a multidimensional time series

**Input:** $p$ maximal homology order, $\Delta$ window size, $s$ stride.
**Data:** A multivariate time series $(Y_t)_{t\in[0,L]} \in \mathbb{R}^D \times [0, L]$

**for** $t$ *in* $[\![0, \lfloor (L-\Delta)/s \rfloor]\!]$ **do**
  compute similarity matrix on the slice $[st, st+\Delta]$, $S_t = 1 - Corr(Y_{[st,st+\Delta]})$;
  compute the Vietoris-Rips filtration for $([\![1, D]\!], E, S_t)$ ;
  **for** *homology dimension* $d$ *in* $[\![0, p-1]\!]$ **do**
    compute order $d$ persistence diagram $X_t^{(d)}$ of the Rips filtration;

**Output:** $p$ (discrete) time series of persistence diagrams $X_t^{(d)}$, $t \in [\![0, \lfloor (L-\Delta)/s \rfloor]\!]$, $d \in [\![0, p-1]\!]$.

The two subsections that follow give details on the different steps of Algorithm 1. We start with a brief description of the TDA tools that we use.

### 2.1.1 Vietoris-Rips persistent homology for weighted graphs

In this subsection we briefly explain how discrete measures are associated to weighted graphs, encoding their multiscale topological structure through persistent homology theory. We refer the reader to Edelsbrunner and Harer (2010); Chazal et al. (2016); Boissonnat et al. (2018) for a general and thorough introduction to persistent homology.

Recall that given a set V, an (abstract) simplicial complex is a set K of finite subsets of V such that $\sigma \in K$ and $\tau \subset \sigma$ implies $\tau \in K$. Each set $\sigma \in K$ is called a simplex of $K$. The dimension of a simplex $\sigma$ is defined as $|\sigma| - 1$ and the dimension of $K$ is the maximum dimension of any of its simplices. Note that a simplicial complex of dimension 1 is a graph. A simplicial complex classically inherits a canonical structure of topological space obtained by representing each simplex by a geometric standard simplex (convex hull of a finite set of affinely independent points in an Euclidean space) and "gluing" the simplices along common faces. A filtered simplicial complex $(K_\alpha)_{\alpha \in I}$, or filtration for short, is a nested family of complexes indexed by a set of real numbers $I \subset \mathbb{R}$: for any $\alpha, \beta \in I$, if $\alpha \leqslant \beta$ then $K_\alpha \subseteq K_\beta$. The parameter $\alpha$ is often seen as a scale parameter.

Let $G$ be a complete non-oriented weighted graph with vertex set $V$ and real valued edge weight function $s : V \times V \to \mathbb{R}$, $(v, v') \mapsto s_{v,v'}$, satisfying $s_{v,v'} := s_{v',v}$ for any pair of vertices $(v, v')$.

**Definition 1** *Let $\alpha_{\min} \leqslant \min_{v,v' \in V} s_{v,v'}$ and $\alpha_{\max} \geqslant \max_{v,v' \in V} s_{v,v'}$ be two real numbers. The Vietoris-Rips filtration associated to $G$ is the filtration $(\mathrm{VR}_\alpha(G))_{\alpha \in [\alpha_{\min}, \alpha_{\max}]}$ with vertex set $V$ defined by*

$$\sigma = [v_0, \cdots, v_k] \in \mathrm{VR}_\alpha(G) \quad \text{if and only if} \quad s_{v_i,v_j} \leqslant \alpha, \quad \text{for all} \quad i, j \in [\![0, k]\!],$$

*for $k > 1$, and $[v] \in \mathrm{VR}_\alpha(G)$ for any $v \in V$ and any $\alpha \in [\alpha_{\min}, \alpha_{\max}]$.*

The topology of $\mathrm{VR}_\alpha(G)$ changes as $\alpha$ increases: existing connected components may merge, loops and cavities may appear and be filled, etc. Persistent homology provides a mathematical framework and efficient algorithms to encode this evolution of the topology (homology) by recording the scale parameters at which topological features appear and disappear. Each such feature is then represented as an interval $[\alpha_b, \alpha_d]$ representing its life span along the filtration. Its length $\alpha_d - \alpha_b$ is called the persistence of the feature. The set of all such intervals corresponding to topological features of a given dimension $d$ - $d = 0$ for connected components, $d = 1$ for 1-dimensional loops, $d = 2$ for 2-dimensional cavities, etc... - is called the persistence barcode of order $d$ of $G$. It is also classically represented as a discrete multiset $D_d(G) \subset [\alpha_{\min}, \alpha_{\max}]^2$ where each interval $[\alpha_b, \alpha_d]$ is represented by the point with coordinates $(\alpha_b, \alpha_d)$ - a basic example is given on Figure 2. Adopting the perspective of Chazal and Divol (2018); Royer et al. (2021); Chazal et al. (2021), in the sequel of the paper, the persistence diagram $D_d(G)$ will be considered as a discrete measure: $D_d(G) := \sum_{p \in D_d(G)} \delta_p$ where $\delta_p$ is the Dirac measure centered at $p$. To control the influence of the possibly many low persistence features, the atoms in the previous sum can be weighted:

$$D_d(G) := \sum_{(b,d) \in D_d(G)} \omega(b, d) \delta_{(b,d)},$$

where $\omega : \mathbb{R}^2 \to \mathbb{R}_+$ may either be a continuous function which is equal to 0 along the diagonal or just a constant renormalization factor equal to the total mass of the diagram. Notice that, in practice, there exist various libraries to efficiently compute persistence diagrams, such as, e.g., The GUDHI Project (2015).

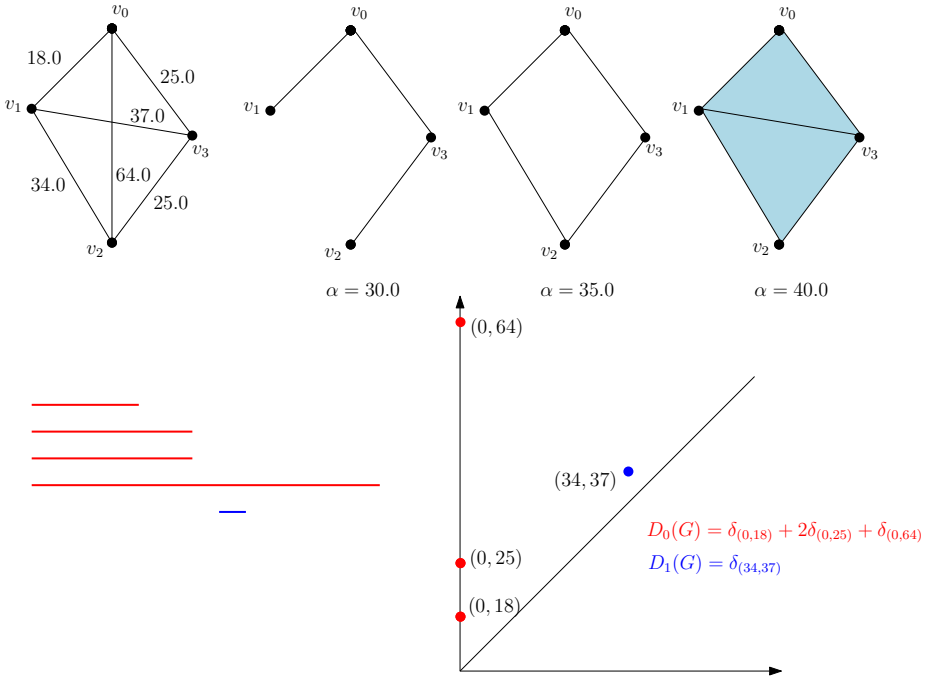


Figure 2: The persistence diagrams of order 0 and 1 of a simple weighted graph $G$ whose vertices are 4 points in the real plane and edge weights are given by the squared distances between them. Here $\alpha_{\min}$ and $\alpha_{\max}$ are chosen to be 0 and 64 respectively. The first line represents $G$ and $\mathrm{VR}_\alpha(G)$ for different values of $\alpha$. The persistence barcodes and diagrams of order 0 and 1 are represented in red and blue respectively on the second line.

The relevance of the above construction relies on the persistence stability theorem Chazal et al. (2016). It ensures that close weighted graphs have close persistence diagrams. More precisely, if $G, G'$ are two weighted graphs with same vertex set $V$and edge weight functions $s : V \times V \to \mathbb{R}$ and $s' : V \times V \to \mathbb{R}$ respectively, then for any order $d$, the so-called bottleneck distance between the persistence diagrams $D_d(G)$ and $D_d(G')$ is upper bounded by $\|s - s'\|_\infty := \sup_{v,v' \in V} |s_{v,v'} - s'_{v,v'}|$ - see Chazal et al. (2014) for formal persistence stability statements for Vietoris-Rips complexes.

### 2.1.2 From similarity matrices to persistence diagrams

Recall here that the data is assumed to be a multivariate time series $(Y_t)_{t\in[0,L]} \in \mathbb{R}^D \times [0, L]$. We intend here to extract the topological information pertaining to the dependence structure between the $D$ channels via persistent homology.

To do so, for a window size $\Delta > 0$ and a stride parameter $s > 0$, we begin by slicing the $D$-dimensional time series into $n + 1$ sub-intervals of the form $[st, st + \Delta]$, where $0 \leqslant t \leqslant n$ and $n = \lfloor (L - \Delta)/s \rfloor$.

Then, for each sub-interval $[st, st + \Delta]$, a coherence graph $G_t$ is built, starting from the fully-connected graph $([\![1, D]\!], E)$ and specifying edge values as $s_{i,j,t} = 1 - \mathrm{Cor}_t(Y_i, Y_j)$, that is 1 minus the correlation between channels $i$ and $j$ computed in the interval $t$.

The size of the time bins $\Delta$ is the time series equivalent of the resolution in images. In practice, choosing a suitable resolution $\Delta$ requires some prior information on the resolution or scale at which the anomalies occur in the time series and might be detected. Up to our knowledge, methods dedicated to detecting changes in time series implicitly use some form of this prerequisite, see Section 4. Another way to formulate this is that the notion of anomalies in time series is better defined *at a certain resolution* $\Delta$, but a proper and explicit definition is outside the scope of this work.

Since the number $n + 1$ of coherence graphs satisfies $n = \lfloor (L - \Delta)/s \rfloor$, choosing a large stride $s$ produces a reduced amount of sample size $n$ for the task that follows. In practice, the stride $s$ is chosen to be small, sometimes 1, as small as the computation time and power allow so as to feed the next step with as much data as possible. From a theoretical point of view, it turns out that the choice of $s$ has no impact on the convergence results (see Section 3.3 for details).

Lastly, the persistence diagrams of the Vietoris-Rips filtration are computed (one per homology order), resulting in sequences of diagrams $X_t^{(d)}$, with $0 \leqslant t \leqslant n$ and $d$ is the homological order. An example of sequences of windows and corresponding persistence diagrams is represented Figure 3. In what follows, a fixed homology order is considered, so that the index $d$ is removed. In practice, the vectorization steps that follows are performed order-wise, as well as the anomaly detection procedure.

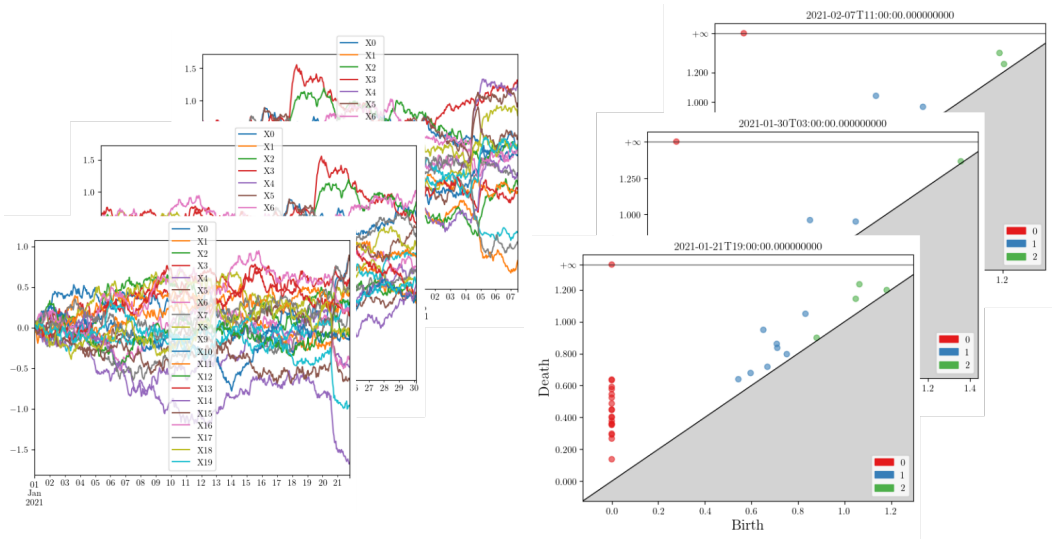


Figure 3: Left: three sliding windows on an illustrative Ornstein–Uhlenbeck (AR1) synthetic process with additive, punctual anomalies. Right: corresponding to those sliding windows, the three topological descriptors (persistence diagrams with Homology dimension 0 (red), 1 (blue) and 2 (green) features), according to Algorithm 1.

It is worth noting that other dependence measures such as the ones based on coherence Ombao and Pinto (2021) may be chosen instead of correlation to build the weighted graphs.

Such alternative choices do not affect the overall methodology nor the theoretical results provided below.

In the numerical experiments, we give results for the correlation weights, that have the advantages of simplicity and carry a few insights: following Bourakna et al. (2022), such weights are enough to detect structural differences in the case where the channels $Y_j$ are mixture of independent components $Z_p$'s, the weights of the mixture being given by a (hidden) graph on the $p$'s whose structure drives the behavior of the observed persistence diagrams.

Finally, it is worth recalling here that Vietoris-Rips filtration may be built on top of arbitrary metric spaces, so that the persistence diagram construction may be performed in more general cases, encompassing valued graphs (with value on nodes or edges) for instance. The vectorization and detection steps below are based on the inputs of such persistence diagrams, and can be applied verbatim to any situation where persistence diagrams can be built from data.

### 2.2 Step 2: convert persistence diagrams into vectors

Once a sequence of persistence diagrams $(X_i)_{i=1,\dots,n}$ is built from data, the next step is to convert these persistence diagrams into a vector-valued sequence. There exists several data-driven methods to perform this vectorization step such as Hofer et al. (2019); Carrière et al. (2020) to name a few supervised ones.

The simplest methods Adams et al. (2017); Royer et al. (2021) consist in evaluating a persistence diagram $X$ considered as a discrete measure on several test functions of the form $u \mapsto \psi(\|u - c\|/\sigma)$, where $\psi$ is a fixed kernel, $c$ and $\sigma$ are respectively a centroid and a scale that vary to provide different test functions. Intuitively speaking, this amounts to encode how much mass a persistence diagram spread around different centers at different scales. Notice that such vectorizations fit within the general frameworks of linear representations of persistence diagrams, see Wu et al. (2024); Divol and Chazal (2019).

In the case where the persistence diagrams $(X_i)_{i=1,\dots,n}$ are i.i.d., it has been shown that the centroids and scales selection procedure exposed in Royer et al. (2021) offers several advantages compared to the fixed grid approach of Persistence Image Adams et al. (2017). For instance, for the same budget $K$ of centroids, the approach of Royer et al. (2021) experimentally outperforms the fixed-grid approach (see Royer et al. 2021, Table 3). In most of the application depicted in Royer et al. (2021); Chazal et al. (2021) as well as in Section 4, a budget $K = 10$ is enough to capture most of the topological information. Such a low-dimensional vectorization usually spares the user a dimension reduction step. Furthermore some theoretical guarantees are available Chazal et al. (2021) that assess the relevance of this procedure in a clustering framework. We thus adopt the same strategy for the dependent case.

**Algorithm 2:** Persistence diagram vectorization

**Input:** $K$: dimension of the output vector. $T$: stopping time.
**Data:** $X_1, \ldots, X_n$ discrete measures.
Use Algorithm 3 (with stopping time $T$) or 4 to get $K$ centroids $\mathbf{c}^{(T)} = (c_1^{(T)}, \ldots, c_K^{(T)})$.
**for** $i = 1, \ldots, n$ **do**
  Use Algorithm 5 with parameter $\mathbf{c}^{(T)}$ on $X_i$ to get $v_i \in \mathbb{R}^K$.
**Output:** Vectorization $\mathbf{v} = (v_1, \ldots, v_n)$.

Let us mention here that this vectorization algorithm may be applied to any dependent sequence of measures (such as texts, point processes realizations, etc.), persistence diagrams data being one example of such a framework. The following subsections give details on the centroids computation algorithms (Algorithm 3 and 4) and the vectorization algorithm (Algorithm 5) in the persistence diagram case.

### 2.2.1 Centroids computation

Recall here from Section 2.1.1 that the persistence diagrams $X_i$'s are thought of as discrete measures on $\mathbb{R}^2$, that is

$$X_i = \sum_{(b,d)\in D_i} \omega_{(b,d)} \delta_{(b,d)},$$

where $D_i$ is the $i$-th persistence diagram considered as a multiset of points (see Section 2.1.1), and $\omega_{(b,d)}$ are weights given to points in the persistence diagram (usually given as a function of the distance from the diagonal, see e.g. Adams et al. 2017).

The goal is to find $K$ centroids $(c_1, \ldots, c_K) \in \mathbb{R}^2$ and scales $(\sigma_1, \ldots, \sigma_k)$ such that the vectorization $X_i \mapsto (X_i(du)\psi(\|u-c_1\|/\sigma_1), \ldots, X_i(du)\psi(\|u-c_K\|/\sigma_K))$ is an accurate enough representation of the original persistence diagrams (where $X(du)f(u)$ means integration of $f$ with respect to $X$).

The key idea behind the ATOL procedure Chazal et al. (2021) is to choose as centroids nearly optimal minimizers of the empirical least-square criterion

$$\mathbf{c} = (c_1, \ldots, c_K) \mapsto \bar{X}_n(du) \min_{j=1,\ldots,k} \|u - c_j\|^2, \tag{1}$$

where $\bar{X}_n$ denotes the empirical mean measure, $\bar{X}_n = \frac{1}{n}\sum_{i=1}^n X_i$.

As mentioned above, this method offers several advantages over fixed-grid strategies such as Persistence Image in the i.i.d. case (see Royer et al. 2021; Chazal et al. 2021 for a more in-depth discussion). The two algorithms that follow expose two methods to approximately minimize (1), their theoretical justification is exposed in Section 3. Let us begin with the batch algorithm.

**Algorithm 3:** Centroids computation - ATOL - Batch algorithm

**Input:** $K$: number of centroids. $T$: stopping time.

**Data:** $X_1, \dots, X_n$ discrete measures.

**Initialization:** $\mathbf{c}^{(0)} = (c_1^{(0)}, \dots, c_K^{(0)})$ randomly chosen from $\bar{X}_n$.

**for** $t = 1, \dots, T$ **do**

  **for** $j = 1, \dots, K$ **do**

    $W_{j,t-1} \leftarrow \{x \in \mathbb{R}^2 \mid \forall i \neq j \quad \|x - c_j^{(t-1)}\| \leqslant \|x - c_i^{(t-1)}\|\}$ (ties arbitrarily broken).

    **if** $\bar{X}_n(W_{j,t-1}) \neq 0$ **then**

      $c_j^{(t)} \leftarrow (\bar{X}_n(du)\left(u \mathbb{1}_{W_{j,t-1}}(u)\right))/\bar{X}_n(W_{j,t-1})$.

    **else**

      $c_j^{(t)} \leftarrow$ random sample from $\bar{X}_n$.

**Output:** Centroids $\mathbf{c}^{(T)} = (c_1^{(T)}, \dots, c_K^{(T)})$.

Algorithm 3 is the same as in the i.i.d. case (Chazal et al., 2021, Algorithm 1). Moreover, almost the same convergence guarantees as in the i.i.d. case may be proven: for a good-enough initialization, only $2\log(n)$ iterations are needed to achieve a statistically optimal convergence (see Theorem 5 below). Therefore, a practical implementation of Algorithm 3 should perform several threads based on different initializations (possibly in parallel), each of them being stopped after $2\log(n)$ steps, yielding a complexity in time of $O(n\log(n) \times n_{start})$ (where $n_{start}$) is the number of threads.

As for the i.i.d. case, an online version of Algorithm 3 may be conceived, based on mini-batches. In what follows, for a convex set $C \subset \mathbb{R}^d$, we let $\pi_C$ denote the Euclidean projection onto $C$.

**Algorithm 4:** Centroids computation - ATOL - Minibatch algorithm

**Input:** $K$: number of centroids. $q$: size of mini-batches. $R$: maximal radius.

**Data:** $X_1, \dots, X_n$ discrete measures.

**Initialization:** $\mathbf{c}^{(0)} = (c_1^{(0)}, \dots, c_K^{(0)})$ randomly chosen from $\bar{X}_n$. Split $X_1, \dots, X_n$ into $n/q$ mini-batches of size $q$:
$B_{1,1}, B_{1,2}, B_{1,3}, B_{1,4}, \dots, B_{t,1}, B_{t,2}, B_{t,3}, B_{t,4}, \dots, B_{T,1}, B_{T,2}, B_{T,3}, B_{T,4}$, $T = n/4q$.

**for** $t = 1, \dots, T$ **do**

  **for** $j = 1, \dots, K$ **do**

    $W_{j,t-1} \leftarrow \{x \in \mathbb{R}^2 \mid \forall i \neq j \quad \|x - c_j^{(t-1)}\| \leqslant \|x - c_i^{(t-1)}\|\}$ (ties arbitrarily broken).

    **if** $\bar{X}_{B_{t,1}}(W_{j,t-1}) \neq 0$ **then**

      $c_j^{(t)} \leftarrow \pi_{\mathcal{B}(0,R)}\left((\bar{X}_{B_{t,3}}(du)\left(u \mathbb{1}_{W_{j,t-1}}(u)\right))/\bar{X}_{B_{t,1}}(W_{j,t-1})\right)$.

    **else**

      $c_j^{(t)} \leftarrow c_j^{(t-1)}$.

**Output:** Centroids $\mathbf{c}^{(T)} = (c_1^{(T)}, \dots, c_K^{(T)})$.

Contrary to Algorithm 3, Algorithm 4 differs from its i.i.d. counterpart given in Chazal et al. (2021). First, the theoretically optimal size of batches is now driven by the decay of the $\beta$-mixing coefficients of the sequence of persistence diagrams, as will be made clear by Theorem 6 below.

Second, half of the sample are wasted (the $B_{t,j}$'s with even $j$). This is due to theoretical constraints to ensure that the mini-batches that are used are spaced enough to guarantee a prescribed amount of independence. Of course, the even $B_{t,j}$'s could be used to compute a parallel set of centroids. However, in the numerical experiments, all the sample are used (no space is left between minibatches) with no noticeable side effect.

From a computational viewpoint, Algorithm 4 is single-pass, so that, if $n_{start}$ threads are run, the global complexity is in $O(n \times n_{start})$.

Figure 4 below depicts an instance of centroids computed by these algorithms.

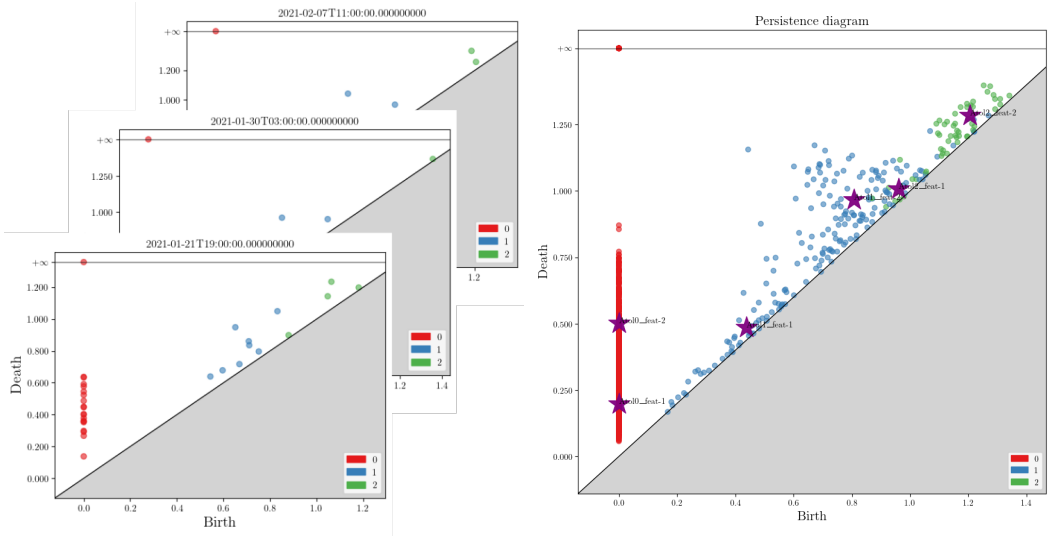


Figure 4: Left: three representative topological descriptors in the form of persistence diagrams with Homology dimensions 0, 1 and 2. Right: sum of topological descriptors and their centroids (stars in purple, two by dimension) computed from them in dimensions 0, 1 and 2 according to Algorithm 3 or Algorithm 4.

### 2.2.2 Conversion into a vector-valued sequence

Once the centroids $\mathbf{c}^{(T)}$ built, the next step is to convert the persistence diagrams $(X_i)_{i=1,\dots,n}$ into vectors. The approach here is the same as in Royer et al. (2021). Denoting by $\psi_{AT} : u \mapsto \exp(-u^2)$, a persistence diagram $X_i$ is mapped onto

$$v_i = \left( X_i(du)\psi_{AT}(\|u - c_1^{(T)}\|/\sigma_1), \dots, X_i(du)\psi_{AT}(\|u - c_K^{(T)}\|/\sigma_K) \right), \tag{2}$$

where the scales $\sigma_j$'s are defined by

$$\sigma_j = \min_{\ell \neq j} \|c_\ell^{(T)} - c_j^{(T)}\|/2, \tag{3}$$

that roughly seizes the width of the area corresponding to the centroid $c_j^{(T)}$. Other choices of kernel $\psi$ are possible (see e.g. Chazal et al. 2021), as well as other methods for choosing the scales. The proposed approach has the benefit of not requiring a careful parameter tuning step, and seems to perform well in practice.

We encapsulate this vectorization method as follows, and an example vectorization is shown in Figure 5.

**Algorithm 5:** Vectorization step

**Input:** Centroids $c_1, \ldots, c_K$
**Data:** A persistence diagram $X$
**for** $j = 1, \ldots, K$ **do**
  Compute $\sigma_j$ as in (3);
  $v_j \leftarrow X(du)\psi_{AT}(\|u - c_j\|/\sigma_j)$
**Output:** Vectorization $v = (v_1, \ldots, v_K)$.

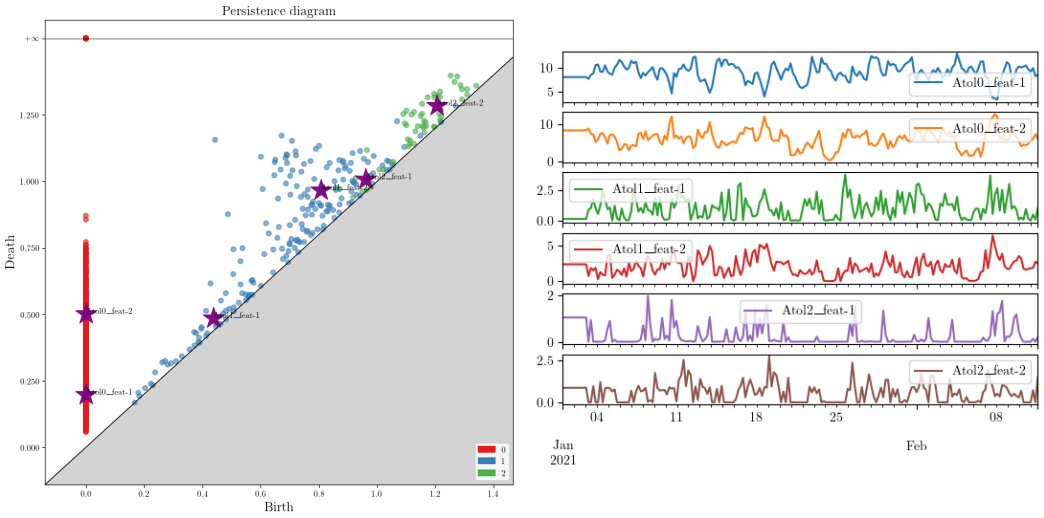


Figure 5: Left: sum of topological descriptors and their centroids (stars in purple, two by dimension) computed from them in dimensions 0, 1 and 2 by Algorithm 3 or 4. Right: the derived topological vectorization of the entire time series computed relative to each center according to Algorithm 5.

### 2.3 Step 3: build an anomaly score

We assume now that we observe the vector-valued sequence $v_1, \ldots, v_n$ of vectorized persistence diagrams, and intend to build a procedure to determine whether a new diagram (processed with Algorithm 2) may be thought of as an anomaly.

We first estimate the 'normal' behavior of the vectorizations $v_i$'s that are thought of as originating from a base regime. Namely, we build the sample means and covariances

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} v_i,$$
$$\hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}(v_i - \hat{\mu})(v_i - \hat{\mu})^T. \tag{4}$$

In the case where the base sample can be corrupted, robust strategies for mean and covariance estimation such as Rousseeuw and Driessen (1999); Hubert et al. (2018) may be

employed. More precisely for a fraction parameter $h \in [0;1]$ we use the Minimum Covariance Determinant estimator (MCD) defined by

$$\begin{aligned}
\hat{I} &\in \underset{I \subset \{1,\dots,n\}, |I| = \lceil nh \rceil}{\arg\min} \operatorname{Det}\left(\frac{1}{|I|}\sum_{i \in I}(v_i - \bar{v}_I)(v_i - \bar{v}_I)^T\right), \\
\hat{\mu} &= \bar{v}_{\hat{I}}, \\
\hat{\Sigma} &= c_0\left(\frac{1}{|\hat{I}|}\sum_{i \in \hat{I}}(v_i - \hat{\mu})(v_i - \hat{\mu})^T\right),
\end{aligned} \tag{5}$$

where $\bar{v}_I$ denotes empirical mean on the subset $I$, and $c_0$ is a normalization constant that can be found in Hubert et al. (2018). In the anomaly detection setting, assuming that at least half of sample points are not anomalies lead to the conservative choice $h = 1/2$. In practice, for $K$-dimensional vectorizations $v_1, \dots, v_n$, we adopt the default value $h = (n + K + 1)/2n$ prescribed in Rousseeuw and Driessen (1999); Lopuhaa and Rousseeuw (1991) that maximizes the finite sample breakdown point of the resulting covariance estimator. In all the experiments exposed in Section 4, we use the approximation of (5) provided in Rousseeuw and Driessen (1999).

Now, for a new vector $v$, a detection score is built via

$$s^2(v) = (v - \hat{\mu})^T \hat{\Sigma}^{-1}(v - \hat{\mu}), \tag{6}$$

that expresses the normalized distance to the mean behavior of the base regime. We refer to an illustrative example in Figure 6.

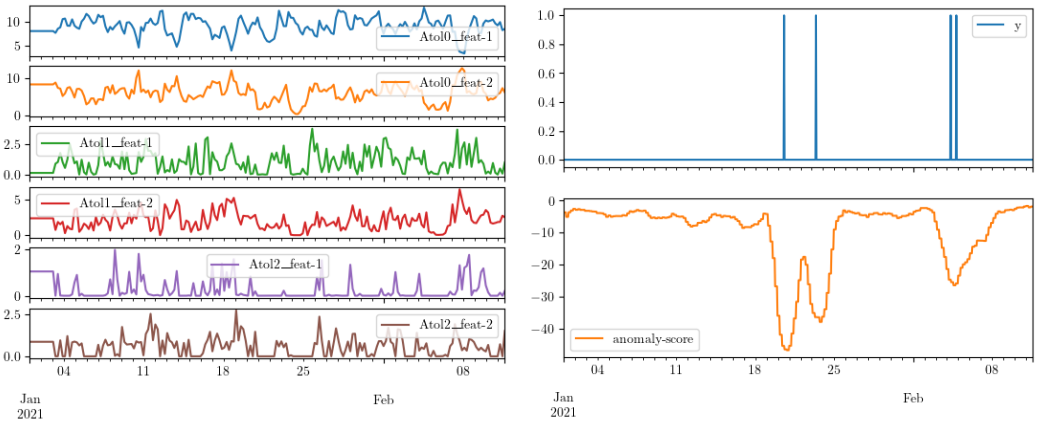


Figure 6: Left: the derived topological vectorization of the entire time series computed relative to each center according to Algorithm 5. Right: (top, in blue) the binary anomalous timestamps $y$ of the original signal, matches (bottom, in orange) the topological anomaly score based on the dimension 0 and 1 features of Algorithm 6.

If we let $\hat{s}$ denote the score function based on data $(Y_t)_{t \in [0,T]}$, then anomaly detection tests of the form

$$T_\alpha(v) = \mathbb{1}_{\hat{s}(v) \geqslant t_\alpha}$$

may be built. To assess the relevance of this family of tests, the ROC_AUC and RANGE_PR_AUC metrics are used in the Application Section 4.

Should a test with specific type I error $\alpha$ be needed, a calibration of $t_\alpha$ as the $1-\alpha$ quantile of scores on a sample from the normal regime could be performed. Section 3.2 theoretically proves that this strategy is grounded.

### 2.4 Summary

We can now summarize the whole procedure into the following algorithm, with complementary descriptive scheme in Figure 1.

**Algorithm 6:** TADA: Detection score from base regime time series

**Input:**

- *TDA parameter:* an integer $p$ maximal homology order;
- *Time series parameters:* a window size $\Delta$ and stride $s$;
- *Vectorization parameters:* a dimension $K$, a number of threads $n_{start}$, and possibly a stopping time $T$ or a mini-batch size $q$;
- *Anomaly score parameters:* a support fraction parameter $h$.

**Data:** A multivariate time series $(Y_t)_{t\in[0,L]} \in \mathbb{R}^D \times [0, L]$ (base regime, possibly corrupted)

1 Convert $(Y_t)_{t\in[0,L]}$ into $n$ persistence diagrams $(X_i)_{i=1,\dots,n}$ via Algorithm 1;
2 Convert $(X_i)_{i=1,\dots,n}$ into $v_1, \dots, v_n$ using Algorithm 2;
3 Compute an anomaly score function $s : \mathbb{R}^K \to \mathbb{R}^+$ defined by (6) from the vector-valued sequence.

**Output:** An anomaly score function $s : \mathbb{R}^K \to \mathbb{R}^+$.

As such Algorithm 6 produces an anomaly score for all persistence diagrams $(X_i)_{i=1,\dots,n}$. For practical uses one often needs to get anomaly scores on the *initial* timestamps $t \in [0, L]$. In this case a standard time-sequence remapping is performed, we defer to Section 4 for explicit details.

We summarize here the impact and defaults choices of the parameters:

- *Time series parameters:* the window size $\Delta$ must be chosen according to the length of the anomalies the user wants to detect. To be conservative, the stride $s$ should be chosen as small as computation time allows, and we use a heuristic default $s = \Delta/10$, although in theory any reasonable choice of $s$ yields the same result.

- *p maximal homology order:* this depends on the complexity of the objects analyzed as each homology order encodes topological features of the corresponding dimension (connected components for homology order 0, cycles and holes for homology order 1, 2-dimensional voids for homology order 2...). The computational cost of persistence and the difficulty to interpret persistence diagrams increase with the homology order. As a consequence, in most practical applications the maximum homology order is set to 1 or 2. In our time series application case, we restrict to $p = 1$ as the produced dependence structure generally do not exhibit relevant higher dimensional features.

- *Vectorization parameters:* the dimension $K$ can be chosen small, as it is one of the virtues of this algorithm that it is very efficient even at very small scales. The default value $K = 5$ by homology dimension yields good results in practice. Since Step 2 and 3 are fast from a computational point of view, the user may try several increasing values of $K$ and stop when satisfied. The number of threads $n_{start}$ corresponds to the number of parallel runs of Algorithm 3 or 4, based on different initializations. The default value $n_{start} = 10$ performs well in practice. The choice of $T$ is relevant only when Algorithm 3 is used, in which case the default value $T = 2\log(n)$ works well in theory and practice. The choice of $q$ is relevant only when Algorithm 4 is used. In this case several increasing values of $q$ may be tried, or $q$ may be chosen based on prior knowledge on the mixing coefficient of the sequence of persistence diagrams (see Section 3.1 for details).

- *Anomaly score parameters:* The default choice $h = (n + K + 1)/2n$ is theoretically grounded (see Rousseeuw and Driessen 1999; Lopuhaa and Rousseeuw 1991) and works well in practice. There are two situations where $h$ could be chosen larger than default. First, when user have access to normal regime data to calibrate the method, in this case robust estimators are not needed and $h$ should be set to 1. Second, in the small sample case, $nh$ may be too small to yield an accurate enough covariance estimator in (5), so that larger values of $h$ may be tried.

Overall, only two parameters play an important role in our procedure: the window size $\Delta$ that corresponds to the resolution of the anomalies to be detected, and $K$ the size of the vectorizations of persistence diagrams. Importantly, no prior knowledge on the temporal dependence structure (the mixing coefficients for instance) of the original time series $(Y_t)_{t\in[0,L]}$ is needed. All the experiments described in Section 4 use the default values.

Our proposed anomaly detection procedure Algorithm 6 has a *lean* design for the following reasons. First it has few parameters, all of them coming with default values, at the exception of the $\Delta$ time series resolution that must be adjusted by reflecting on the type of anomalies one is looking for. Second, very little tuning is needed. In the entire application sections to come, the only parameter to change will be that resolution parameter $\Delta$, a parameter shared with other methods. All other parameters are set to default values. Third, upon learning some data, TADA does not require a lot of memory: only the results of Algorithms 5 (centroids) and 6 (training vectorization mean and variance) are needed in order to produce topological anomaly scores. This implies that our methodology is easy to deploy, and requires no memory of training data which is often welcome in contexts of privacy for instance. It also means that the methodology is able to compare very favorably to methods that are memory-heavy such as tree-based methods, neural networks, etc.

## 3 Theoretical results

In this section we intend to assess the relevance of our methodology from a theoretical point of view. Sections 3.1 and 3.2 give results in the general case where the sample is a stationary sequence of random measures. Section 3.3 provides some details on how persistence diagrams built from multivariate time series as exposed in Section 2.1.1 can be casted into this general framework.

### 3.1 Convergence of Algorithms 3 and 4

In what follows we assume that $X_1, \dots, X_n$ is a stationary sequence of random measures over $\mathbb{R}^d$, with common distribution $X$. Some assumptions on $X$ are needed to ensure convergence of Algorithms 3 and 4.

First, let us introduce here $\mathcal{M}_{N_{\max}}(R, M)$ as the set of random measures that are bounded in space, mass and support size.

**Definition 2** *For $R, M > 0$ and $N_{\max} \in \mathbb{N}^*$, we let $\mathcal{M}_{N_{\max}}(R, M)$ denote the set of discrete measures $\mu$ on $\mathbb{R}^d$ that satisfy*

1. $\mathrm{Supp}(\mu) \subset \mathcal{B}(0, R)$,
2. $\mu(\mathbb{R}^2) \leqslant M$,
3. $|\mathrm{Supp}(\mu)| \leqslant N_{\max}$.

*Accordingly, we let $\mathcal{M}(R, M)$ denote the set of measures such that 1 and 2 hold.*

With a slight abuse, if $X$ denotes a distribution of random measures, we will write $X \in \mathcal{M}_{N_{\max}}(R, M)$ whenever $X \in \mathcal{M}_{N_{\max}}(R, M)$ almost surely. As detailed in Section 3.3, persistence diagrams built from correlation matrices satisfy the requirements of Definition 2.

In the i.i.d. case, Chazal et al. (2021) proves that the output of Algorithm 3 with $T = 2\log(n)$ iterations returns a statistically optimal approximation of

$$\mathbf{c}^* \in \mathcal{C}_{opt} = \underset{\mathbf{c} \in (\mathbb{R}^2)^K}{\arg\min} \, \mathbb{E}(X)(du) \min_{j=1,\dots,K} \|u - c_j\|^2 := F(\mathbf{c}), \tag{7}$$

where $\mathbb{E}(X)$ is the so-called mean measure $\mathbb{E}(X) : A \in \mathcal{B}(\mathbb{R}^2) \mapsto \mathbb{E}(X(A))$. Note here that $X \in \mathcal{M}(R, M)$ ensures that $\mathcal{C}_{opt}$ is non-empty (see, e.g., Chazal et al. 2021, Section 3). For the aforementioned result to hold, a structural condition on $\mathbb{E}(X)$ is also needed.

For a vector of centroids $\mathbf{c} = (c_1, \dots, c_k) \in \mathcal{B}(0, R)^k$, we let

$$\begin{aligned} W_j(\mathbf{c}) = \{x \in \mathbb{R}^d \mid \forall i < j \quad \|x - c_j\| < \|x - c_i\| \quad \text{and} \\ \forall i > j \quad \|x - c_j\| \leqslant \|x - c_i\|\}, \\ N(\mathbf{c}) = \{x \mid \exists i < j \quad x \in W_i(\mathbf{c}) \quad \text{and} \quad \|x - c_i\| = \|x - c_j\|\}, \end{aligned}$$

so that $(W_1(\mathbf{c}), \dots, W_k(\mathbf{c}))$ forms a partition of $\mathbb{R}^2$ and $N(\mathbf{c})$ represents the skeleton of the Voronoi diagram associated with $\mathbf{c}$. The margin condition below requires that the mass of $\mathbb{E}(X)$ around $N(\mathbf{c}^*)$ is controlled, for every possible optimal $\mathbf{c}^* \in \mathcal{C}_{opt}$. To this aim, let us denote by $\mathcal{B}(A, t)$ the $t$-neighborhood of $A$, that is $\{y \in \mathbb{R}^d \mid \mathrm{d}(y, A) \leqslant t\}$, for any $A \subset \mathbb{R}^d$ and $t \geqslant 0$. The margin condition then writes as follows.

**Definition 3** *$\mathbb{E}(X) \in \mathcal{M}(R, M)$ satisfies a margin condition with radius $r_0 > 0$ if and only if, for all $0 \leqslant t \leqslant r_0$,*

$$\sup_{\mathbf{c}^* \in \mathcal{C}_{opt}} \mathbb{E}(X)\left(\mathcal{B}(N(\mathbf{c}^*), t)\right) \leqslant \frac{B p_{min}}{128 R^2} t,$$

*where $\mathcal{B}(N(\mathbf{c}^*), t)$ denotes the $t$-neighborhood of $N(\mathbf{c}^*)$ and*

1. $B = \inf_{\mathbf{c}^* \in \mathcal{C}_{opt}, j \neq i} \|c_i^* - c_j^*\|$,
2. $p_{min} = \inf_{\mathbf{c}^* \in \mathcal{C}_{opt}, j=1,\dots,k} \mathbb{E}(X)\,(W_j(\mathbf{c}^*))$.

According to (Chazal et al., 2021, Proposition 7), $B$ and $p_{\min}$ are positive quantities whenever $\mathbb{E}(X) \in \mathcal{M}(R, M)$. In a nutshell, a margin condition ensures that the mean distribution $\mathbb{E}(X)$ is well-concentrated around $k$ poles. For instance, finitely-supported distributions satisfy a margin condition. Up to our knowledge, margin-like conditions are always required to guarantee convergence of Lloyd-type algorithms Tang and Monteleoni (2016); Levrard (2018) in the i.i.d. case.

For our motivating example of persistence diagrams built from a sequence of correlation matrices between a time series channels, we cannot assume anymore independence between observations. To adapt the argument of Chazal et al. (2021) in this framework, a quantification of dependence between discrete measures is needed. We choose here to seize dependence between observations via $\beta$-mixing coefficients, whose definition is recalled below.

**Definition 4** *For $t \in \mathbb{Z}$ we denote by $\sigma(-\infty, t)$ (resp. $\sigma(t, +\infty)$) the sigma-fields generated by $\dots, X_{t-1}, X_t$ (resp. $X_t, X_{t+1} \dots$). The beta-mixing coefficient of order $q$ is then defined by*

$$\beta(q) = \sup_{t \in \mathbb{Z}} \mathbb{E}\left[ \sup_{B \in \sigma(t+q, +\infty)} |\mathbb{P}(B \mid \sigma(-\infty, t)) - \mathbb{P}(B)| \right].$$

Recalling that the sequence of persistence diagrams is assumed to be stationary, its beta-mixing coefficient of order $q$ may be subsequently written as

$$\beta(q) = \mathbb{E}(\mathrm{d}_{TV}(P_{(X_q, X_{q+1}, \dots) | \sigma(\dots, X_0)}, P_{(X_q, X_{q+1}, \dots)})),$$

where $\mathrm{d}_{TV}$ denotes the total variation distance and $P_Z$ denotes the distribution of $Z$, for a generic random variable $Z$. As detailed in Section 3.3, mixing coefficients of persistence diagrams built from a multivariate time series may be bounded in terms of mixing coefficients of the base time series. Whenever these coefficients are controlled, results from the i.i.d. case may be adjusted to the dependent one.

We begin with an adaptation of (Chazal et al., 2021, Theorem 9) to the dependent case.

**Theorem 5** *Assume that $X_1, \dots, X_n$ is stationary, with distribution $X \in \mathcal{M}_{N_{max}}(R, M)$, for some $N_{max} \in \mathbb{N}^*$. Assume that $\mathbb{E}(X)$ satisfies a margin condition with radius $r_0$, and denote by $R_0 = \frac{Br_0}{16\sqrt{2}R}$, $\kappa_0 = \frac{R_0}{R}$. For $q \in \mathbb{N}^*$, choose $T \geqslant \lceil \frac{\log(n/q)}{\log(4/3)} \rceil$, and let $\mathbf{c}^{(T)}$ denote the output of Algorithm 3.*

*If $q$ is such that $\beta(q)^2/q^3 \leqslant n^{-3}$, and $\mathbf{c}^{(0)} \in \mathcal{B}(\mathcal{C}_{opt}, R_0)$, then, for $n$ large enough, with probability larger than $1 - c\frac{qkM^2}{n\kappa_0^2 p_{\min}^2} - 2e^{-x}$, we have*

$$\inf_{\mathbf{c}^* \in \mathcal{C}_{opt}} \|\mathbf{c}^{(T)} - \mathbf{c}^*\|^2 \leqslant \frac{B^2 r_0^2}{512 R^2} \left(\frac{q}{n}\right) + C \frac{M^2 R^2 k^2 d \log(k)}{p_{min}^2} \left(\frac{q}{n}\right)(1 + x),$$

*for all $x > 0$, where $C$ is a constant.*

*Moreover, if $q$ is such that $\beta(q)/q \leqslant n^{-1}$ and $\mathbf{c}^{(0)} \in \mathcal{B}(\mathbf{c}^*, R_0)$, it holds*

$$\mathbb{E}\left( \inf_{\mathbf{c}^* \in \mathcal{C}_{opt}} \|\mathbf{c}^{(T)} - \mathbf{c}^*\|^2 \right) \leqslant C \frac{dk^2 R^2 M^2 \log(k)}{\kappa_0^2 p_{\min}^2} \left(\frac{q}{n}\right)$$

Intuitively speaking, Theorem 5 provides the same guarantees as in the i.i.d. case, but for a 'useful' sample size $n/q$. This 'useful' sample size corresponds to the number of sample measures that are spaced enough (in fact $q$-spaced) so that they may be considered independent enough (with respect to the targeted convergence rate in $q/n$). This point of view seems ubiquitous in machine learning results based on dependent sample (see, e.g., Agarwal and Duchi 2013, Theorem 1 or Mohri and Rostamizadeh 2010, Lemma 7).

Assessing the optimality of the requirements on $\beta(q)$ is difficult. Following (Mohri and Rostamizadeh, 2010, Corollary 20) and comments below, the $\beta(q) \leqslant q/n$ condition we require to get a convergence rate in expectation seems optimal for polynomial decays ($\beta(q) = O(q^{-a})$, $a > 0$) in an empirical risk minimization framework. However, this choice leads to a convergence rate in $(q/n)^{(a-1)/(4a)}$ for Mohri and Rostamizadeh (2010), larger than our $(q/n)$ rate. Though the output of Algorithm 3 is not an empirical risk minimizer, it is likely that it has the same convergence rate as if it were (based on a similar behavior for the plain $k$-means case, see e.g., Levrard 2018). The difference between convergence rates given in Mohri and Rostamizadeh (2010) and Theorem 5 might be due to the fact that Mohri and Rostamizadeh (2010) settles in a 'slow rate' framework, where the convexity of the excess risk function is not leveraged, whereas a local convexity result is a key argument in our result (explicited by Chazal et al. 2021, Lemma 21).

In a fast rate setting (i.e. when the risk function is strictly convex), (Agarwal and Duchi, 2013, Theorem 5) also suggests that a milder requirement in $\beta(q)/q \leqslant n^{-1}$ might be enough to get a $O(q/n)$ convergence rate in expectation, for online algorithms under some assumptions that will be discussed below Theorem 6 (convergence rates for an online version of Algorithm 3). Up to our knowledge there is no lower bound in the case of stationary sequences with controlled $\beta$ coefficients that could back theoretical optimality of such procedures.

At last, the sub-exponential rate we obtain in the deviation bound under the stronger condition $\beta(q)^2/q^3 \leqslant n^{-3}$ seems better than the results proposed in (Mohri and Rostamizadeh, 2010, Corollary 20) or (Agarwal and Duchi, 2013, Theorem 5) in terms of large deviations (in $(q/n)x$ here to get an exponential decay). Determining whether the same kind of result may hold under the condition $\beta(q) \leqslant q/n$ remains an open question, as far as we know.

Nonetheless, Theorem 5 provides some convergence rates (in expectation) for several decay scenarii on $\beta(q)$:

- if $\beta(q) \leqslant C\rho^q$, for $\rho < 1$, then an optimal choice of $q$ is $q = c\log(n)$, providing the same convergence rate as in the i.i.d. case (Chazal et al., 2021, Theorem 9), up to a $\log(n)$ factor.

- if $\beta(q) = Cq^{-a}$, for $a > 0$, then an optimal choice of $q$ is $q = Cn^{\frac{1}{a+1}}$, that yields a convergence rate in $n^{-1+\frac{1}{a+1}}$.

In the last case, letting $a \to +\infty$ allows to retrieve the i.i.d. case, whereas $a \to 0$ has for limiting case the framework when only one sample is observed (thus leading to a non-learning situation).

Whatever the situation, a benefit of Algorithm 3 is that a correct choice of $q$, thus the prior knowledge of $\beta(q)$, is not required to get an at least consistent set of centroids, by

choosing $T = \lceil \frac{\log(n)}{\log(4/3)} \rceil$. This will not be the case for the convergence of Algorithm 4, where the size of minibatches $q$ is driven by prior knowledge on $\beta$.

**Theorem 6** *Let $q$ be large enough so that $\frac{\beta(q/18)}{q^2} \leqslant n^{-2}$ and $q \geqslant c_0 \frac{k^2M^2}{p_{\min}^2 \kappa_0^2} \log(n)$, for a constant $c_0$ that only depends on $\int_0^1 \beta^{-1}(u)du$. Provided that $\mathbb{E}(X)$ satisfies a margin condition, if the initialization satisfies the same requirements as in Theorem 5, then the output of Algorithm 4 satisfies*

$$\mathbb{E}\left( \inf_{\mathbf{c}^* \in \mathcal{C}_{opt}} \|\mathbf{c}^{(T)} - \mathbf{c}^*\|^2 \right) \leqslant 128 \frac{kdMR^2}{p_{\min}(n/q)}.$$

As in Rio (1993), the generalized inverse $\beta^{-1}$ is defined by $\beta^{-1}(u) = |\{k \in \mathbb{N}^* \mid \beta(k) > u\}|$. In particular, for $\beta(q) \sim q^{-a}$, $\int_0^1 \beta^{-1}(u)du$ is finite only if $a > 1$ (that precludes the asymptotic $a \to 0$).

The requirement $\beta(q)/q^2 = O(n^{-2})$ is stronger than in Theorem 5, thus stronger that the $\beta(q)/q = O(n^{-1})$ suggested by (Agarwal and Duchi, 2013, Theorem 5) in a similar online setting. Note however that for (Agarwal and Duchi, 2013, Theorem 5) to provide a $O(q/n)$ rate under the requirement $\beta(q)/q = O(n^{-1})$, two other terms have to be controlled:

1. a total step sizes term in $\sum_{t=1}^T \|\mathbf{c}^t - \mathbf{c}^{(t-1)}\|$ that must be of order $O(1)$. Controlling this term would require an slight adaptation of Algorithm 4, for instance by clipping gradients.

2. a regret term in $\mathbb{E}\left( \sum_{t=1}^T \bar{X}_{B_T}(du) \left[ \mathrm{d}^2(u, \mathbf{c}^{(t)}) - \mathrm{d}^2(u, \mathbf{c}^*) \right] \right)$ that must be of order $O(q)$. The behavior of this term remains unknown in our setting, so that determining whether the milder condition $\beta(q)/q = O(n^{-1})$ is sufficient remains an open question.

Let us emphasize here that, to optimize the bound in Theorem 6, that is to choose the smallest possible $q$, prior knowledge of $\beta(q)$ is required. This can be the case when the original multivariate time series $Y_t$ follows a recursive equation as in Bourakna et al. (2022). Otherwise, these coefficients may be estimated, using histograms as in McDonald et al. (2015) for instance.

As in the batch case, the required lower bound on $q$ corresponds to the 'optimal' choice of minibatch spacings so that consecutive even minibatches may be considered as i.i.d.. It is then not a surprise that we recover the same rate as in the i.i.d. case, but with $n/q$ samples (see Chazal et al. 2021, Theorem 10). As for the batch situation, several decay scenarii may be considered:

- for $\beta(q) \leqslant C\rho^q$, $\rho < 1$, choosing $q = c_0 \frac{k^2M^2}{p_{\min}^2 \kappa_0^2} \log(n)$ for a large enough $c_0$ is enough to satisfy the requirements of Theorem 6, and yields the same result as in the i.i.d. case ((Chazal et al., 2021, Theorem 10)).

- for $\beta(q) = Cq^{-a}$, $\beta^{-1}(u) = (C/u)^{1/a}$. An optimal choice for $q$ is then $Cn^{\frac{2}{a+2}}$, leading to a convergence rate in $n^{-1+\frac{2}{a+2}}$.

Let us mention here that the stronger condition in Theorem 6 leads to a slower convergence bound for the polynomial decay case, compared to the output of Algorithm 3. Again, assessing optimality of exposed convergence rates remains an open question, up to our knowledge.

### 3.2 Test with controlled type I error rate

In this section, we investigate the type I error of the test

$$T_\alpha \mapsto \mathbb{1}_{s(v)\geqslant t_{n,\alpha}},$$

where $s$ is the score function built in Section 2.3 and $t_{n,\alpha}$ will be built from sample to achieve a type I error rate below $\alpha$.

To keep it simple, we assume that $\Sigma$ and $\mu$ in (4) are computed from a separate sample, so that we observe

$$\tilde{v}_i = \Sigma^{-1/2}(v_i - \mu)$$

from a stationary sequence of measures, resulting in a stationary sequence of vectors. Whether $\Sigma$ and $\mu$ should be computed on the same sample, extra terms involving concentration of $\Sigma$ and $\mu$ around their expectations should be added, as in the i.i.d. case.

We let $Z$ denote the common distribution of the $s(v_i) = \|\tilde{v}_i\|$'s, that represent the 'normal' behavior distribution of the time series structure. For the test $T_\alpha$ introduced above, its type I error is then

$$\mathbb{P}_{\|\tilde{v}\|\sim Z}(\|\tilde{v}\| > t_{n,\alpha}).$$

A common strategy here is to choose a $t_{n,\alpha}$ from sample, such as

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\|\tilde{v}_i\|>t_{n,\alpha}} \leqslant \alpha - \delta,$$

for a suitable $\delta < \alpha$. In what follows we denote by $\hat{t}$ such an empirical choice of threshold. The following result ensures that this natural strategy remains valid in a dependent framework.

**Proposition 7** *Let $q \in [\![1, n]\!]$, and $\alpha$, $\delta$ be positive quantities that satisfy*

$$5\sqrt{\alpha}\sqrt{\frac{\log(n)}{(n/q)}} \leqslant \delta < \alpha.$$

*If $\hat{t}$ is chosen such that*

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\|\tilde{v}_i\|>\hat{t}} \leqslant \alpha - \delta,$$

*then, with probability larger than $1 - \frac{4q}{n} - \beta(q)\sqrt{\frac{n}{\alpha q}}$, it holds*

$$\mathbb{P}_{\|\tilde{v}\|\sim Y}(\|\tilde{v}\| > \hat{t}) \leqslant \alpha.$$

In other words, Proposition 7 ensures that the anomaly detection test $\mathbb{1}_{\|\tilde{v}\|\geqslant\hat{t}}$ has a type I error below $\alpha$, with high probability. Roughly, this bound ensures that, for confidence levels $\alpha$ above the statistical uncertainty of order $q/n$, tests with the prescribed confidence level may be achieved via increasing the threshold by a term of order $\sqrt{\alpha q/n}$.

As for Theorem 5 or 6, choosing the smaller $q$ that achieves $\beta(q)^2/q^3 \leqslant \alpha/n^3$ optimizes the probability bound in Proposition 7:

- for $\beta(q) \leqslant C\rho^q$, $q$ of order $C' \log(n)$ is enough to satisfy $\beta(q)/q^2 \leqslant n^{-2}$, providing the same results as in the i.i.d. case, up to a $\log(n)$ factor.

- for $\beta(q) = Cq^{-a}$, an optimal choice for $q$ is of order $n^{\frac{3}{2a+3}}\alpha^{-1/(2a+3)}$, that leads to the same bounds as in the i.i.d. case, but with useful sample size $n/q = n^{\frac{2a}{2a+3}}\alpha^{1/(2a+3)}$.

Although this bound might be sub-optimal, it can provides some pessimistic prescriptions to select a threshold $\alpha - \delta$, provided that the useful sample size $n/q$ is known. For instance, assuming $\log(n) \leqslant 6$, for $\alpha = 5\%$ the minimal $\delta$ is of order $2.7/\sqrt{n/q}$, that is negligible with respect to $\alpha$ whenever $n/q$ is large compared to roughly 3000.

### 3.3 Theoretical guarantees for persistence diagrams built from multivariate time series

We discuss here how the outputs $(X_t)_{t=1,\dots n}$ of Algorithm 1, based on a $D$-dimensional time series $(Y_t)_{t\in[0;L]}$, fall in the scope of the previous sections. Recall that a persistence diagram may be thought of as a discrete measure on $\mathbb{R}^2$ (see Section 2.2.1). In a nutshell, if $(Y_t)_{t\in[0;L]}$ is stationary with a certain profile of mixing coefficients, then so would $(X_t)_{t=1,\dots n}$.

*Stationarity*: Since, for $t = 1, \dots, n$, $X_t$ may be expressed as $f((Y_u)_{u\in[s(t-1);s(t-1)+\Delta]}$, for some function $f$, then stationarity of $(X_t)_{t=1,\dots,n}$ follows from stationarity of $(Y_t)_{t\in[0;L]}$.

*Boundedness*: We intend to prove that the outputs of Algorithm 1 are in the scope of Definition 2. Let $d$ be an homology dimension, $t \in [\![1; n]\!]$, and recall that $X_t$ is the order $d$ persistence diagram built from the Vietoris-Rips filtration of $([\![1; D]\!], E, S_t)$, where $E$ is the set of all edges and $S_t = 1 - Corr(Y_{[s(t-1);s(t-1)+\Delta]})$ gives the weights that are filtered (see Section 2.1.1). First note that, for every $1 \leqslant i, j \leqslant D$, $S_{t,(i,j)} \in [0; 2]$, so that every point in the persistence diagram is in $[0; 2]^2$. Next, since a birth of a $d$-order feature is implied by an addition of a $d$-order simplex in the filtration (see for instance Boissonnat et al. (2018) Section 11.5, Algorithm 11), the total number of points in the diagram is bounded by $\binom{D}{d+1}$. At last, for a bounded weight function $\omega$, the total mass of $X_t$ may be bounded by $\binom{D}{d+1} \times \|\omega\|_\infty$. We deduce that $X_t \in \mathcal{M}_{N_{\max}}(R, M)$ (Definition 2), with $R \leqslant 4$, $N_{\max} \leqslant \binom{D}{d+1}$, and $M \leqslant \binom{D}{d+1} \times \|\omega\|_\infty$. Note that in the experiments, we set $\omega \equiv 1$.

*Mixing coefficients*: Here we expose how the mixing coefficients of $(X_t)_{t=1,\dots,n}$ (Definition 4) may be bounded in terms of those of $(Y_t)_{t\in[0;L]}$. Let us denote these coefficients by $\beta$ and $\tilde{\beta}$. If the stride $s$ is larger than the window size $\Delta$, then it is immediate that, for all $q \geqslant 1$, $\beta(q) \leqslant \tilde{\beta}(qs - \Delta)$. If the stride $s$ is smaller (or equal) than $\Delta$, then, denoting by $q_0 = \lfloor(\Delta/s)\rfloor + 1$, we have, for $q < q_0$, $\beta(q) \leqslant 1$ (overlapping windows), and, for $q \geqslant q_0$, $\beta(q) \leqslant \tilde{\beta}(qs - \Delta)$. The mixing coefficients of $X_t$ may thus be controlled in terms of those of $Y_t$. For fixed $\Delta$ and $s$, this ensures that mixing coefficients of $X_t$ and $Y_t$ have the same profile (and leads to the same convergence rates in Theorem 5 and 6).

- If $\tilde{\beta}(q) \leqslant C_Y q^{-a}$, for $C_Y, a > 0$, then, for any $q \geqslant q_0$, $\beta(q) \leqslant C_Y\,(s - \Delta/q_0)^{-a}\,q^{-a}$, so that $\beta(q) \leqslant C_X(qs)^{-a}$, for some constant $C_X$ (depending on $q_0$ and $a$).

- If $\tilde{\beta}(q) \leqslant C_Y\tilde{\rho}^q$, for $C_Y > 0$ and $\tilde{\rho} < 1$, then, for any $q \geqslant q_0$, $\beta(q) \leqslant C_Y(\tilde{\rho}^{(s-(\Delta/q_0))})^q$, so that $\beta(q) \leqslant C_X\rho^{qs}$, for some $C_X > 0$ and $\rho < 1$ depending on $q_0$ and $\tilde{\rho}$.

In turn, mixing coefficients of $Y_t$ may be known or bounded, for instance in the case where it follows a recursive equation (see, e.g., Pham and Tran 1985, Theorem 3.1), or inferred (see, e.g., McDonald et al. 2015). Interestingly, the topological wheels example provided in Section 4.1 (borrowed from Bourakna et al. 2022) falls into the sub-exponential decay case.

This relation between the mixing coefficients of $(Y_t)_{t\in[0;L]}$ and those of $(X_t)_{t=1,\dots,n}$ allows to shed some light on the influence of the stride parameter $s$ on the convergence results. Assume for simplicity that the original time series is discrete with $\tilde{n}$ points and $\Delta$ is fixed. In the two examples above it holds, for $q \geqslant \lfloor \Delta \rfloor + 1$, $\beta(q) \leqslant C\tilde{\beta}(qs)$, where $C$ depends on $\Delta$ and the parameters of $\tilde{\beta}$ only. For a choice of stride $s$, the resulting sample size is $n = \tilde{n}/s$, so that an optimal choice of $q \geq q_0$ with respect to Theorem 5 should satisfy

$$\frac{\beta(q)}{q} = \frac{1}{n} \quad \Leftrightarrow \quad C\frac{\tilde{\beta}(qs)}{(qs)} = \frac{1}{\tilde{n}},$$

to provide a convergence rate in $q/n = (qs)/\tilde{n}$. Let $\tilde{q}_n$ be such that $\tilde{\beta}(\tilde{q}_n) = \tilde{q}_n/\tilde{n}$ (optimal choice of $q$ w.r.t. $\tilde{\beta}$). Then, provided $s$ is not too large so that $\tilde{q}_n/s \geq q_0$, an optimal choice of $q$ w.r.t. the above bounds on $\beta$ is $q_n = \tilde{q}_n/s$, leading to a convergence rate in $\tilde{q}_n/\tilde{n}$, whatever the chosen $s$. This backs the intuition that any reasonable choice of stride $s$ (not too large) should lead to the same theoretical guarantees.

*Margin condition*: The only point that cannot be theoretically assessed in general for the outputs of Algorithm 1 is to know whether $\mathbb{E}(X)$ satisfies the margin condition exposed in Definition 3. As explained below Definition 3, a margin condition holds whenever $\mathbb{E}(X)$ is concentrated enough around $k$ poles. Thus, structural assumptions on $1 - Corr(Y_{[0;\Delta]})$ (for instance $k$ prominent loops) might entail $\mathbb{E}(X)$ to fulfill the desired assumptions (as in Levrard 2015 for Gaussian mixtures). However, we strongly believe that the requirements of Definition 3 are too strong, and that convergence of Algorithms 3 and 4 may be assessed with milder smoothness assumptions on $\mathbb{E}(X)$. This falls beyond the scope of this paper, and is left for future work. The experimental Section 4 to follow assesses the validity of our algorithms in practice.

## 4 Applications

In order to make the case for the efficiency of our proposed anomaly detection procedure TADA, we now present an assortment of both real-case and synthetic applications. The first application (introduced as the Topological Wheels problem) is directly derived from Bourakna et al. (2022). It consists of a synthetic data set designed to mimic dependence patterns in brain signals, and allows to demonstrate the relevance of a topologically based anomaly detection procedure on such complex data. The second application is an up-to-date replication of a benchmark with the TimeEval library from Schmidl et al. (2022) on a large array of synthetic data sets to quantitatively demonstrate competitiveness of the proposed procedure with current state-of-the-art methods. The third application is a real-case data set from Jacob et al. (2021-07) consisting of data traces from repeated executions of large-scale stream processing jobs on an Apache Spark cluster. Lastly we produce interpretability elements for the anomaly detection procedure TADA.

Evaluation of an anomaly detection procedure in the context of time series data has many pitfalls and can be hard to navigate, we refer to the survey of Sørbø and Ruocco

(2023). Here we mainly evaluate anomaly scores with the robustified version of the Area Under the PR-Curve: the Range PR_AUC metric of Paparrizos et al. (2022a) (later just 'RANGE_PR_AUC'), where a metric of 1 indicates a perfect anomaly score, and a metric close to 0 indicates that the anomaly score simply does not point to the anomalies in the data set. For the sake of comparison with the literature we also include the Area Under the ROC-Curve metric (later just 'ROC_AUC'), although this metric is less accurate and powerful in the unbalanced context of anomaly detection (Sørbø and Ruocco 2023). Therefore each collection of anomaly detection problems will yield evaluation statistics, and to summarize comparisons between algorithms we use a critical difference diagram, that is a statistical test between paired populations using the package Herbold (2020). Furthermore we introduce two other statistical summaries of interest:

- the '# $\geqslant$ .9' metric, that is the number of anomaly detection problems for which an algorithm has a RANGE_PR_AUC over .9. A RANGE_PR_AUC over .9 roughly indicates that the algorithm 'finds well' the anomalies in the data set or 'solves' the problem;

- the '#rank1' metric, that is the number of problems for which an algorithm reaches the best RANGE_PR_AUC score over other algorithms, ties being shared.

For the purpose of comparison with the state-of-the-art we draw methods from the recent benchmark of Schmidl et al. (2022). We take the *three best performing* methods from the unsupervised, multivariate category: the 'KMeansAD' anomaly detection based on k-means cluster centers distance using ideas from Yairi et al. (2001), the baseline density estimator k-nearest-neighbors algorithm on time series subsequences 'SubKNN', and 'TorskAD' from Heim and Avery (2019), a modified echo state network for anomaly detection.

Furthermore, in order to better understand the value of the introduced topological methodology, we compare to a close-resembling method that couples the topological features of Algorithm 5 to the isolation forest algorithm from Liu et al. (2008), resulting in an *unsupervised* anomaly detection method denominated as 'Atol-IF' in reference to the Royer et al. (2021) paper. We also couple those topological features to a random forest classifier Breiman (2001-10), resulting in a *supervised* anomaly detection method denominated as 'Atol-RF' that gives an idea on what can be achieved in the supervised context. Lastly, to investigate the differences between the proposed topological analysis and a more standard spectral analysis, we compute spectral features on the correlation graphs coupled to either an isolation forest or to a random forest classifier, in an *unsupervised* anomaly detection method denominated as 'Spectral-IF' and a *supervised* one named 'Spectral-RF'.

In practice all those methods involve a form of time-delay-embedding or subsequence analysis or context window analysis (we use these terms synonymously in this work), that requires to compute a prediction from a window size number $\Delta$ of past observations. $\Delta$ is a key value that acts as the equivalent of image resolution or scale in the domain of time series. In using a subsequence analysis, given a $\Delta$-uplet of timestamps $[t] := (t_1, t_2, ..., t_\Delta)$, once an anomaly score $s_{[t]}$ is produced it is related to that particular $\Delta$-uplet but does not refer to a specific time step. A window reversing step is needed to map the scores to the original timestamps. For fair comparison, we will provide all methods with the following (same) last-step window reversing procedure: for every time step $t$, one computes the sum of windows containing this time step $\hat{s}_t := \sum_{[t']:t\in[t']} s_{[t']}$. Here we choose not to use the more

classical average $\tilde{s}_t := \sum_{[t']:t\in[t']} s_{[t']} / \left(\sum_{[t']:t\in[t']} 1\right)$, since this average produces undesirable border effects (the timestamps at the beginning and end of the signal are contained in less windows, making them over-meaningful after averaging). Using the sum instead has no effect on anomaly scoring (outside of borders) as the metrics are scale-invariant.

For the specific use of TADA in this section, the centroids computation part of Section 2.2.1 is made using $\omega_{(b,d)} = 1$ and computed with the batch version described in Algorithm 3. Our implementation relies on The GUDHI Project (2015) for the topological data analysis part, but also makes use of the Pedregosa et al. (2011) Scikit-learn library for the anomaly detection part, minimum covariance determinant estimation and overall pipelining. The code is published as part of the ConfianceAI program `https://catalog.confiance.ai/` and can be found in the catalog: `https://catalog.confiance.ai/records/4fx8n-6t612`. All computations were made on a standard laptop (i5-7440HQ 2.80 GHz CPU).

### 4.1 Introducing the Topological Wheels data set.

In this first application section we introduce a hard, multiple time series unsupervised problem that emulates brain functions, and compare our method with state-of-the-art anomaly detection methods as well as supervised concurrent methods.

Bourakna et al. (2022) introduces ideas for evaluating methodologies relying on TDA such as ours. They allow to produce a multiple time series with a given node dependence structure from a mixture of latent autoregressive process of order two (AR2). One direct application for this type of data generation is to emulate the network structure of the brain, whose normal connectivity is affected by conditions such as ADHD or Alzheimer's disease. Therefore, in accordance with Bourakna et al. (2022), we design and introduce the Topological Wheels problem: a multiple time series data sets with a latent dependence structure 'type I' as a normal mode, and sometimes a 'type II' latent dependence structure as an abnormal mode. For the type I dependence structure we use a single wheel where every node are connected by pair, and every pair are connected to two others forming a wheel; then we connect a pair of pairs forming an 8 shape or double wheel (see Figure 7). For the type II structure we start from a double wheel and add another connection between two pairs. The first mode of dependence is the prominent mode for the time series duration, and is replaced for a short period at a random time by the second mode of dependence. The total signal involves 64 time series sampled at 500 Hz for a duration of 20 seconds, see Figure 7. We produce ten such data sets and call them the Topological Wheels problem. It consists in being able to detect the change in underlying patterns without supervision. We note that by design the two modes are similar in their spectral profile, so detecting anomalies should be hard for methods that do not capture the overall topology of the dependence structure. The data sets are available through the public The GUDHI Project (2015) library at `github.com/GUDHI/gudhi-data`.

For assessing performances we use a cross-validation like procedure with a focus on evaluation: we perform ten experiments and for each experiment, every method is fitted on one data set and evaluated on the other nine data sets. We then rotate the training data set until all ten data sets have been used for training. We use this particular setup in order to be able to compare supervised and unsupervised methods on comparable grounds. As for method calibration we note the following: all methods are given (and use) the real length

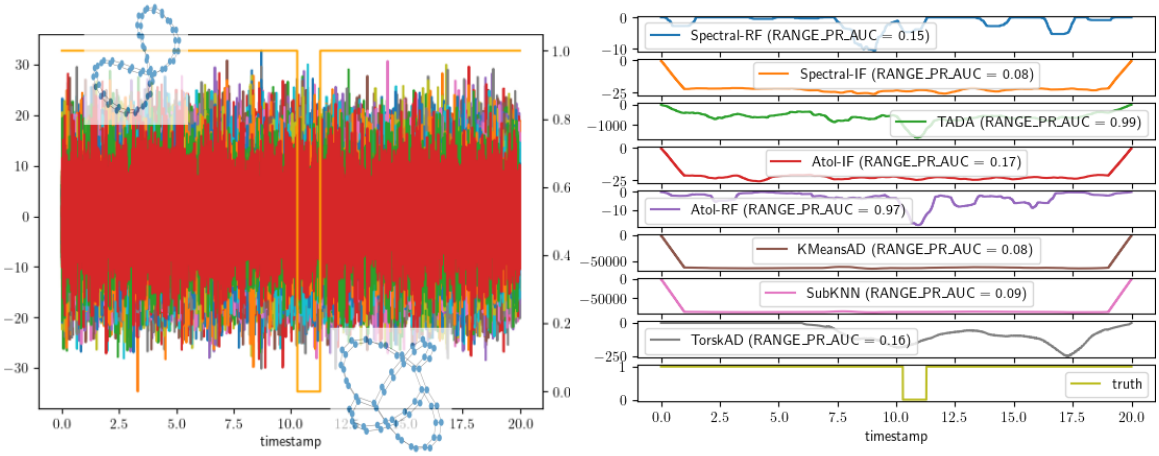


Figure 7: Left: Synthetised time series and the latent generating process (orange) indicating normal connection (double circular wheel with middle connection, on top) or abnormal connection (double circular wheel with two connections, on the bottom). Right: Anomaly scores of all tested methods on one of the data sets, and their RANGE_PR_AUC metric on comparing with the truth (bottom row) in parenthesis.

of the anomalous segment of $\Delta = 500$ consecutive timestamps, and since all of them use window subsequences we set the same stride for all to be $s = \Delta/10 = 50$. Lastly for all methods that use a fixed-size embedding (Atol-based methods and spectral-based methods), we set the support size to $K = 10$.

We first show in Figure 7 the results of one iteration of learning, that is when all methods are trained on a topological wheels data set and evaluated on another. The last row of the figure with label 'truth' shows the underlying signal value of the evaluated data set. The other rows are the computed anomaly score of each method along the time x-axis, with the convention that the lower the score, the more abnormal the signal. The corresponding RANGE_PR_AUC score of each method is written in the label. This first example confirms the intuition that methods that do not rely on topology, that is the spectral method, the k-nearest-neighbor method and the modified echo state network method all fail to capture the anomaly. This is particularly striking for the spectral method as it was trained with supervision. On the other hand all methods based on the topological features manage to capture some indication that there is anomaly in the signal. For the isolation forest method, even though it clearly separates the anomalous segment from the rest, it is not reliable as it seems to indicate other anomalies when there aren't. The random forest supervised method perfectly discriminates the anomalous segment from the rest of the time series, and so does our method almost as reliably.

We now look at the aggregated results for the entire problem, see Figure 8 and times Table 1. We present both ROC_AUC and RANGE_PR_AUC averages with their standard deviations over experiments, as well as the computation times for the sake of completeness. Neither the spectral procedure nor the echo state network, subKNN or k-nearest-neighbor methods are able to capture any information from the Topological Wheels problem. Using topological features with an isolation forest yields competitive results but it is simply inferior

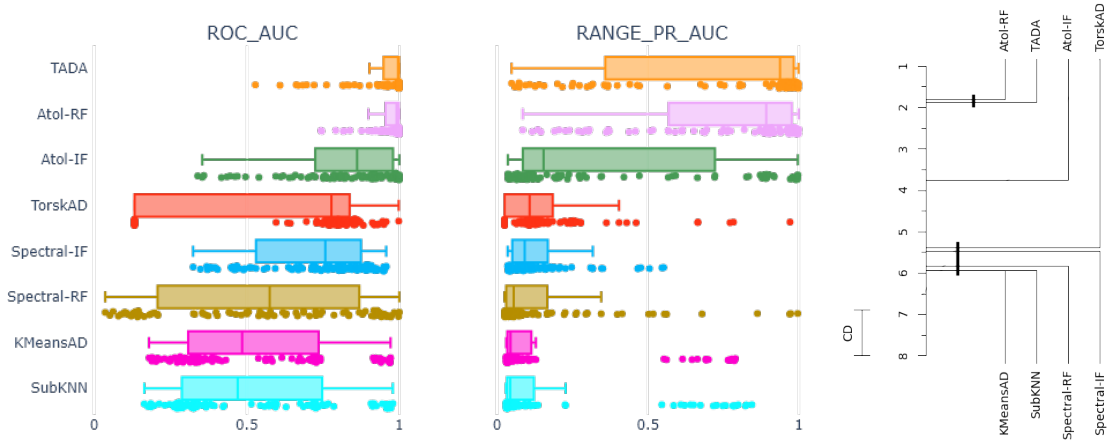


Figure 8: Left: aggregated results for the Topological Wheels problem in the form of box plots for the ROC_AUC and RANGE_PR_AUC metrics, and below them the points the metrics have been computed from - where each point represents a metric score from comparing an anomaly score to the underlying truth. Right: autorank summary (see Herbold (2020)) ranking of methods on the Topological Wheels problem, showing competitiveness between our unsupervised TADA method and the equivalent topological supervised method.

| algorithm | #xp | # $\geqslant .9$ | #rank1 | med time | iqr time |
|---|---|---|---|---|---|
| Atol-IF | 90 | 14 | 8 | 18.2 | 5.4 |
| Atol-RF | 90 | 38 | 43 | 22.3 | 2.5 |
| KMeansAD | 90 | 0 | 0 | 0.6 | 0.0 |
| Spectral-IF | 90 | 0 | 0 | 1.2 | 0.1 |
| Spectral-RF | 90 | 3 | 2 | 1.1 | 0.1 |
| SubKNN | 90 | 0 | 0 | 1.3 | 0.1 |
| TADA | 90 | 51 | 37 | 18.5 | 3.7 |
| TorskAD | 90 | 1 | 2 | 112.8 | 2.0 |

Table 1: Summary statistics on the Topological Wheels problem for the algorithms evaluated. All methods could produce scores for the 90 experiments, and without surprise the methods relying on topological analysis are overwhelmingly dominating other methods in 86 out of 90 experiments. Our unsupervised method TADA is on par with a supervised learning method for the number of problem where it has the best PR_AUC score (#rank1 column). In seconds, the computation median time ('med time') and interquartile range ('iqr time') are standard with respect to the data sizes - also note that computations are not optimized, and in fact performed on a single laptop.

to our procedure. This demonstrates that the key information to this problem lies in the topological embedding which is not surprising, by design. Our procedure solves this problem almost perfectly, and although it is unsupervised it is as competitive as a comparable

supervised method. This experiment demonstrates the impact of topology-based methods for anomaly detection, as the non-topology methods fail to capture any of the signal in the data sets. Our proposed TADA method is clearly the best suited for learning anomalies in this setup.

### 4.2 A benchmark using the TimeEval library.

We now look at a broader and more general array of problems, to evaluate the competitiveness of our method with respect to state-of-the-art methods. For that purpose, we use the GutenTAG multivariate data sets, drawn from Wenig et al. (2022).

We chose the GutenTAG data sets [1] for the ability to generate a great (1174) number of varied anomaly detection problems; they are mostly formed from an insertion of anomalies of various lengths into a multivariate time series of 10 000 timestamps. These anomalies can be abnormal in terms of frequency, variance, or extremum values.

We believe that the GutenTAG data sets are a collection of mostly easy anomaly detection problems, but that they become challenging and interesting considered as a whole. Because they have various types of anomalies encoded in them, evaluating methods with little to no calibration on the entire collection is informative for evaluating performance and robustness in detecting anomalies in a somewhat general case. As the anomalies in these data sets vary from one to a thousand consecutive timestamps, we set the window sizes of the anomaly detectors to be a fixed $\Delta = 100$. Other than that, all other parameters from the previous section are left unchanged (with the exception of the stride that depends on $\Delta$, we keep $s = \Delta/10 = 10$).

| algorithm | #xp | # $\geqslant .9$ | #rank1 | med time | iqr time |
|---|---|---|---|---|---|
| Atol-IF | 1174 | 859 | 373 | 3.8 | 0.4 |
| KMeansAD | 1174 | 383 | 233 | 0.8 | 0.1 |
| Spectral-IF | 1174 | 750 | 302 | 1.7 | 0.2 |
| SubKNN | 1174 | 970 | 996 | 0.9 | 0.3 |
| TADA | 1174 | 530 | 231 | 4.6 | 0.5 |
| TorskAD | 733 | 70 | 21 | 14.4 | 1.3 |

Table 2: Summary statistics on the GutenTAG problem set for the algorithms evaluated. Even though it is ranked fourth by the statistical pairwise-ranking in Figure 9, our unsupervised method TADA is able to solve roughly half of the problems, and is competitive on 231 of them. The other topological method Atol-IF ranks second and is able to solve 859 of the 1174 problems. In seconds, the computation median time ('med time') and interquartile range ('iqr time') are standard with respect to the data sizes involved - computations are performed on a single laptop. TorskAD failed to return scores on a number of data sets due to unknown bugs.

1. After publication the GutenTAG authors have commented that they had an unwanted artifact in the GutenTAG data sets generation (we refer to `https://timeeval.github.io/evaluation-paper/notebooks/Datasets.html`). Therefore we used the new GutenTAG data sets provided by the authors here: `https://github.com/TimeEval/GutenTAG/blob/main/generation_configs/multivariate-test-cases.py` and only remove the instances where the number of time series $D = 500$ caused exceedingly large computing times.

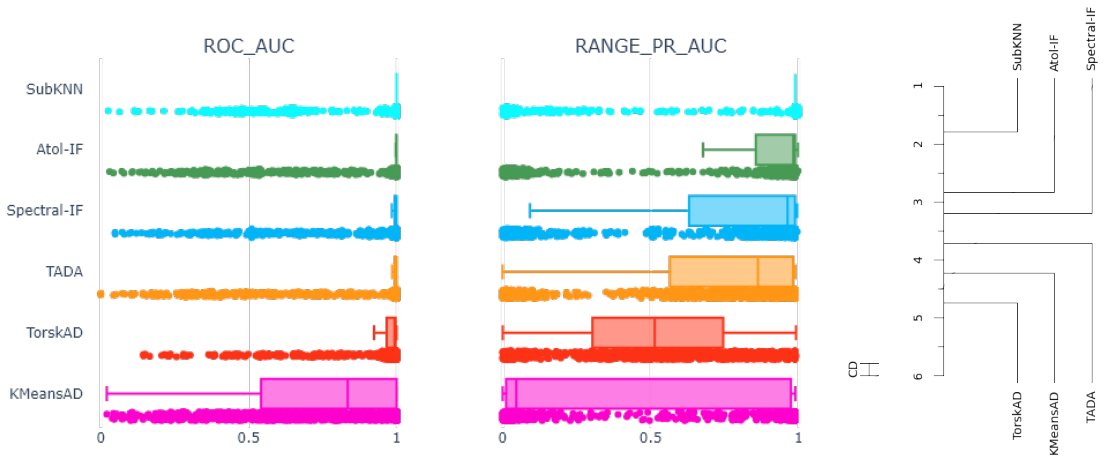


Figure 9: Left: aggregated results for the TimeEval 1084 GutenTAG synthetic data sets in the form of box plots for the ROC_AUC and RANGE_PR_AUC metrics, and below them the points the metrics have been computed from - where each point represents a metric score from comparing an anomaly score to the underlying truth. Right: autorank summary (see Herbold (2020)) ranking of methods, showing fourth place ranking for our purely topological TADA method.

Statistical summaries and results of the experiment on the synthetic data sets are shown Figure 9, and in Table 2. As a reminder, the SubKNN, KMeansAD and TorskAD methods were the top three performing methods from the largest anomaly detection benchmark to date (see Table 3 from Schmidl et al. 2022). Our TADA procedure manages to solve roughly half of the problems and is a top contender among competitors for about a quarter of them. Atol-IF performs better than TADA in this instance, which is not surprising as isolation forest retains much more information from training than TADA, which also implies heavier memory loads. Overall SubKNN is able to perform the best on those data sets, and TADA and Atol-IF show good performances, and in some instances only the topological methods manage to solve the problem, see for instance Figure 11. These results demonstrate competitiveness of our methodology in the unsupervised anomaly detection learning context.

### 4.3 Exathlon real data sets

Lastly we turn to a real collection of data sets: the 15 Exathlon data sets from Jacob et al. (2021-07) consisting in data traces from repeated executions of large-scale stream processing jobs on an Apache Spark cluster, and anomalies are intentional disturbances of those jobs.

Using the same metrics, collection of anomaly detection methods and exact same calibration as in the previous TimeEval experiment, we produce the following results. The main statistical summaries are presented in Figure 12 and Table 3. Overall the topological methods are strong competitors for these data sets, with TADA coming off as the most often number one ranked method. Due to the real nature of the data sets, it is not surprising that the studied methods do not 'solve' them as well as they solve the GutenTAG data sets or the


Figure 10: Correlations between algorithm RANGE_PR_AUC scores on the GutenTAG data set collections. The two topological methods TADA and Atol-IF correlate at .86, and the highest correlation reaches .87 for Atol-IF and Spectral-IF. This indicates that the algorithms involved each solve different GutenTAG problems although some overlap naturally occurs.

TopologicalWheels data sets. We show in Figure 13 the one instance where TADA is able to solve the problem completely, and highlight that no calibration has been needed to do so.

One drawback of the topological methods appearing here is the high variance in execution time, which originates from computing topological features on a great number of sensors. Since our implementation of Algorithm 6 is naive, we point that there are strategies for optimizing computation times: ripser, subsampling, clustering that makes sense, etc. Those strategies are outside the scope of this paper.

### 4.4 Score interpretability

The anomaly score we introduce is constructed from compressing the mean measure of persistence diagrams into $K$ centroids, and analyzing the resulting embedding's main distribution features. Once these centers are learned it is possible to engineer anomaly scores respective to a particular center, or possibly to a set of centers e.g. centers associated with

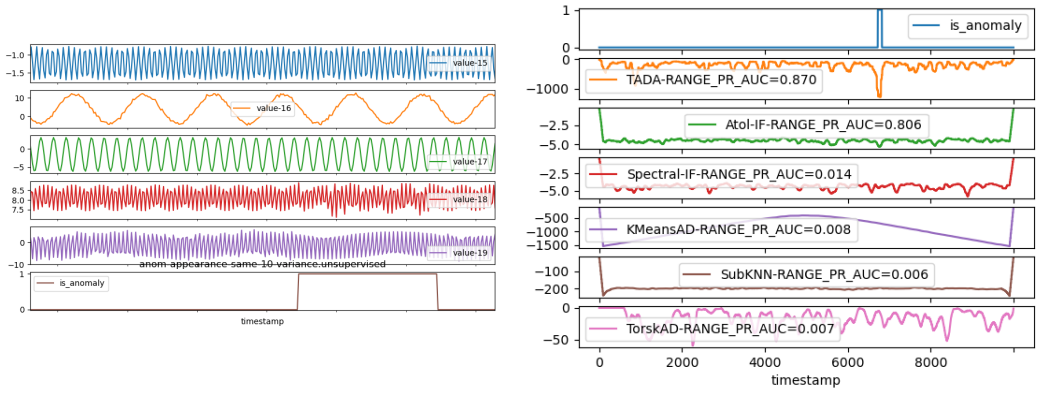


Figure 11: Left: zoom in on a GutenTAG problem instance (sensors selected for visualization) with a variance anomaly for the last two sensors (purple and red), and the ground truth (last row). Right: problem ground truth (top row) and the corresponding anomaly scores of each method with their RANGE_PR_AUC metric in parenthesis. The topological methods TADA and Atol-IF get a good metric score and manage to find the anomalies, while no other method does.

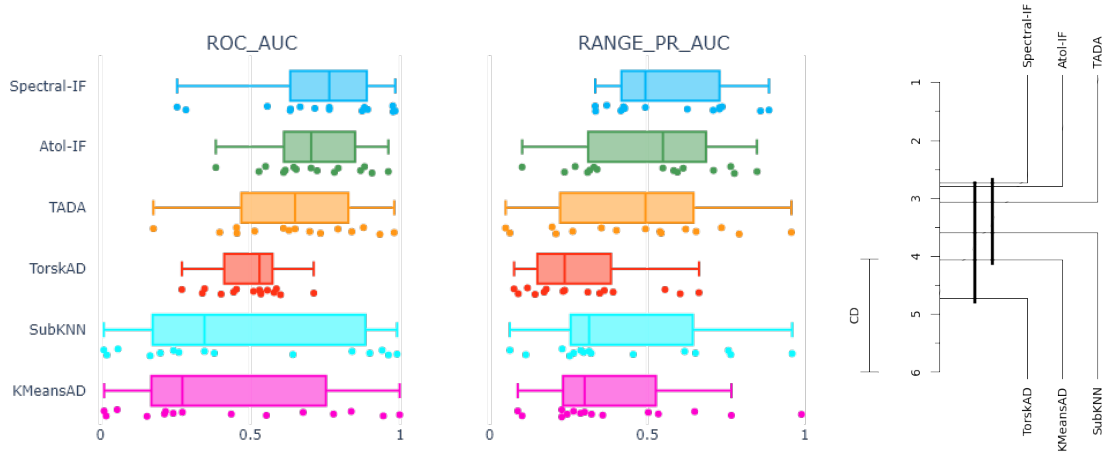


Figure 12: Left: aggregated results for the 35 Exathlon real data sets in the form of box plots for the ROC_AUC and RANGE_PR_AUC metrics, and below them the points the metrics have been computed from - where each point represents a metric score from comparing an anomaly score to the underlying truth. Right: autorank summary (see Herbold (2020)) ranking of methods on the real data sets, showing second and third place ranking for the topological methods Atol-IF and TADA.

a particular homology dimension. Let us examine this first possibility, and introduce the center-targeted scores:

$$\tilde{s}_i = \hat{\Sigma}_{ii}^{-1/2} |v_i - \hat{\mu}_i|, \tag{8}$$

| algorithm | #xp | # ⩾ .9 | #rank1 | med time | iqr time |
|---|---|---|---|---|---|
| Atol-IF | 15 | 0 | 3 | 10.0 | 448.3 |
| KMeansAD | 15 | 1 | 1 | 3.3 | 2.6 |
| Spectral-IF | 15 | 0 | 4 | 6.2 | 1.5 |
| SubKNN | 15 | 1 | 2 | 8.6 | 13.4 |
| TADA | 15 | 1 | 5 | 11.5 | 450.7 |
| TorskAD | 15 | 0 | 0 | 34.3 | 37.4 |

Table 3: Summary statistics on the Exathlon real data problems for the algorithms evaluated. Even though it is ranked third by statistical ranking in Figure 12, our unsupervised method TADA is the top RANGE_PR_AUC score (#rank1 column) over all problems, which hints that it is able to solve different sorts of anomaly detection problems than the others. In seconds, the computation median time ('med time') and interquartile range ('iqr time') are high for the topological methods, see commentaries in the text. Computations are performed on a single laptop.

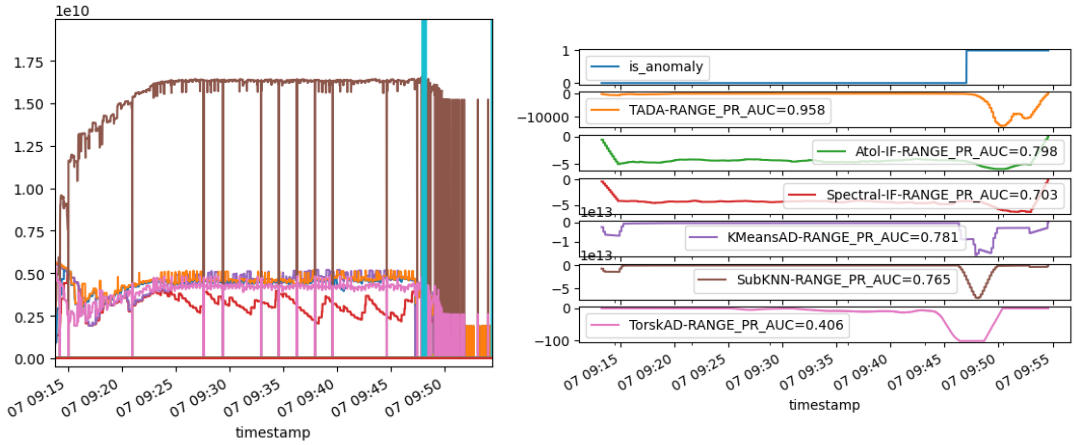


Figure 13: Left: zoom in on data set 3_2_1000000_71. Right: ground truth (top row) and anomaly scores for the six methods and their RANGE_PR_AUC score in parenthesis. While all methods are able to locate the beginning of the anomaly period, only TADA manages to catch it in its entirety.

where $\hat{\mu}, \hat{\Sigma}$ are the estimated mean and covariance of the vectorization $v$ of Algorithm 5 (time indices are implied and omitted for this discussion). These scores can be interpreted as testing for anomalies with respect to a single embedding dimension, as if the vectorization had independent components. These center-targeted scores allow to analyze an original anomaly by looking at the score deviations of each vector component. Because the vector components are integrated from a learned centroid, the scores can be traced back to a specific region in $\mathbb{R}^2$, see for instance Figure 14.

This leads to valuable interpretation. For instance if an abnormal score of TADA were caused by a large deviation in a homology dimension 1 center-related component, it is likely that at that time an abnormal dependence cycle is created for a longer or shorter period of

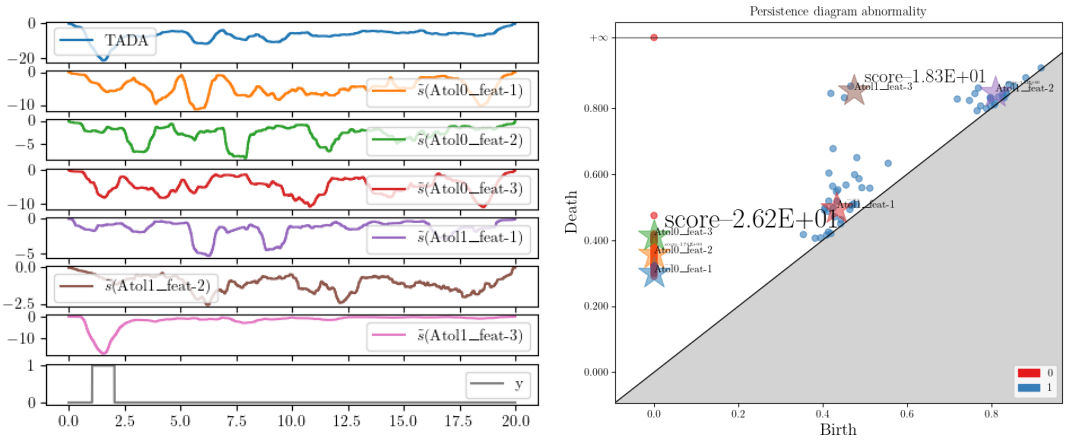


Figure 14: Left: Overall score (top curve, in blue), scores for $\tilde{s}_i$ (middle curves, three top ones for features corresponding to homology dimension 0, three bot ones for features corresponding to homology dimension 1) and ground truth (bottom curve) on a single Topological Wheels data set. The last homology dimension 1 has the strongest correspondence to the overall score, and matches the underlying truth almost exactly. Right: $\tilde{s}_i$ scores of an abnormal persistence diagram on this data set, next to the associated quantization centers (colored stars) of Algorithm 3. The scores are written in a font size proportional to them, so that the more abnormal scores appear bigger. In this instance the quantization centers in dimension 0 and 1 with the highest persistence react to this diagram, hinting at a change in the latent data structure as the highest persistence diagram points are usually associated with signal in comparison with points nearer the diagonal.

time than for the rest of the time series, therefore that the dependence pattern has globally changed in that period of time. See for instance an illustration on the Topological Wheels problem in Figure 14 where globally changing the dependence pattern between sensors is exactly how the abnormal data was produced. In the case where the produced score is deemed abnormal due to a shift in several dimension 0 centers-related components, this indicates an anomaly in the connectivity dependence structure that does not affect the higher order homological features. In this case, the anomaly could be attributed to a default (such as a breakdown) in one of the original sensors for instance.

## 5 Conclusion

It is common knowledge that no anomaly detection method can help with identifying all kinds of anomalies. The framework introduced in this paper is relevant for detecting abnormal topological changes in the global structure of data.

For the motivating example of detecting changes in the dependence patterns between channels of a multivariate time series, our method turns out to be competitive with respect to other state-of-the-art approaches on various benchmark data sets. Naturally, there are many different sorts of anomalies that the proposed method is not able to detect at all. For instance, since the topological embedding is invariant to graph isomorphism, any anomalies linked to node permutation (change of node labeling) cannot be caught. The same is

true for homothetic *jumps*: when signals would simultaneously get identically multiplied, the correlation-based similarity matrix would remain unchanged, leading to unchanged topological embedding. While such invariances are generally thought of as limitative, they can also be a welcome feature if the considered problem introduces the same limitation - for instance when there is labeling uncertainties in sensors collecting data.

The topological anomaly detection finds anomalies that other methods do not seem to discover. It is generally understood that topological information is a form of global information that is complementary to the information gathered by more traditional approaches, e.g. spectral detectors. While confirming this, the above numerical experiment also suggest that topological information is commonly present in various real or synthetic data sets. Therefore for practical applied purposes it is probably best to use our method in combination with other dedicated methods, for instance one that focuses on 'local' data shifts such as the SubKNN method.

Focusing on the case of detection of anomalies in the dependence structure of multivariate time series, it appears that the only parameter that requires a careful tuning in our method is the window size (or temporal resolution) $\Delta$, as it probably also does for most of existing procedures (see Section 4). Designing methods to empirically tune this window size, or to combine the outputs of our method at different resolutions would be a relevant addition to our work, that is left for future research.

Finally, let us recall here the flexibility of the framework we introduce. First, it is not tied to detect changes in correlation structures in time series: we may use Algorithm 1 with other dissimilarity measures between channels, and more generally the two last steps may apply verbatim whenever a sequence of persistence diagrams may be built from data (for instance arising from filtrations on meshed shapes, images, graphs etc.). Second, the vectorization we propose with Algorithm 3 and 5 does not necessarily takes a sequence of persistence diagrams as input: any sequence of measures may be vectorized the same way. It may find applications in monitoring sequences of point processes realizations, as in evolution of distributions of species for instance - see, e.g., Renner et al. (2015). Finally, one may process the output of the vectorization procedure in other ways than building an anomaly score. For instance, using these vectorizations as inputs of any neural network, or change-points detection procedures such as KCP (Arlot et al. 2019) could provide a dedicated method to retrieve change points of a global structure.

## 6 Acknowledgements

This work has been supported by the French government under the 'France 2030' program, as part of the SystemX Technological Research Institute within the Confiance.ai project. The authors also thank the ANR TopAI chair in Artificial Intelligence (ANR–19–CHIA–0001) for financial support.

The authors are grateful to Bernard Delyon for valuable comments and suggestions concerning convergence rates in $\beta$-mixing cases.

## 7 Proofs

Most of the proposed results are adaptations of proofs in the independent case to the dependent one. A peculiar interest lies in concentration results in this framework, we list important ones in the following section.

### 7.1 Probabilistic tools for $\beta$-mixing concentration

In the derivations to follow extensive use will be made of a consequence of Berbee's Lemma.

**Lemma 8** *(Doukhan et al., 1995, Proposition 2)* Let $(X_i)_{i\geqslant 1}$ be a sequence of random variables taking their vales in a Polish space $\mathcal{X}$, and, for $j>0$, denote by

$$b_j = \mathbb{E}\left[\sup_{B\in\sigma(j+1,+\infty)} |\mathbb{P}(B \mid \sigma(-\infty,j)) - \mathbb{P}(B)|\right].$$

Then there exists a sequence $(\tilde{X}_i)_{i\geqslant 1}$ of independent random variables such that, for any $i\geqslant 1$, $\tilde{X}_i$ and $X_i$ have the same distribution and $\mathbb{P}(X_i \neq \tilde{X}_i) \leqslant b_i$.

The above lemma allows to translate standard concentration bounds from the i.i.d. framework to the dependent case, where dependence is seized in terms of $\beta$-mixing coefficients.

Let us recall here the general definition of $\beta$-mixing coefficients from Definition 4. For a sequence of random variables $(Z_t)_{t\in\mathbb{Z}}$ (not assumed stationary), the *beta-mixing* coefficient of order $q$ is

$$\beta(q) = \sup_{t\in\mathbb{Z}} \mathbb{E}\left[\sup_{B\in\sigma(t+q,+\infty)} |\mathbb{P}(B \mid \sigma(-\infty,t)) - \mathbb{P}(B)|\right].$$

If the sequence $(Z_t)_{t\in\mathbb{Z}}$ is assumed to be stationary, $\beta(q)$ may be written as

$$\beta(q) = \mathbb{E}(\mathrm{d}_{TV}(P_{(X_q,X_{q+1},\ldots)|\sigma(\ldots,X_0)}, P_{(X_q,X_{q+1},\ldots)})),$$

where $\mathrm{d}_{TV}$ denotes the total variation distance. We will make use of the following adaptation of Bernstein's inequality to the dependent case.

**Theorem 9** *(Doukhan, 1994, Theorem 4) Let $(X_t)_{t\in\mathbb{Z}}$ be a sequence of (real) variables with $\beta$-mixing coefficients $(\beta(q))_{q\in\mathbb{N}^*}$, that satisfies*

1. $\forall t\in\mathbb{Z} \quad \mathbb{E}(X_t)=0$,
2. $\forall t,n\in\mathbb{Z}\times\mathbb{N} \quad \mathbb{E}\left|\sum_{j=1}^n X_{t+j}\right|^2 \leqslant n\sigma^2$,
3. $\forall t \quad |X_t|\leqslant M$ *a.s.*.

*Then, for every $x\geqslant 0$,*

$$\mathbb{P}\left(\frac{1}{n}\left|\sum_{t=1}^n X_t\right| \geqslant 2\sigma\sqrt{\frac{x}{n}} + \frac{4Mx}{3n}\right) \leqslant 4e^{-x} + 2\beta\left(\lceil\frac{n}{17}\rceil - 1\right).$$

To apply Theorem 9, a bound on the variance term is needed. Such bounds are available in the stationary case under slightly milder assumptions (see, e.g., Rio 1993). For our purpose, a straightforward application of (Rio, 1993, Theorem 1.2, a)) will be sufficient, exposed below.

**Lemma 10** *Let $X_t$ denote a centered and stationary sequence of real variables with $\beta$-mixing coefficients $(\beta(q))_{q\in\mathbb{N}^*}$, such that $|X_t| \leqslant M$ a.s..*

*Then it holds*

$$\frac{1}{n}\mathbb{E}\left(\sum_{j=1}^{n} X_j\right)^2 \leqslant 4M^2 \int_0^1 \beta^{-1}(u)du,$$

*where $\beta^{-1}(u) = \sum_{k\in\mathbb{N}} \mathbb{1}_{\beta(k)>u}$.*

### 7.2 Proofs for Section 3

This section gathers the proofs our theoretical results: convergence of the algorithms (Theorem 5 and 6) and type I error control of the subsequent testing procedure (Proposition 7).

#### 7.2.1 Proof of Theorem 5

**Proof** [Proof of Theorem 5]

We begin by the proof of Theorem 5. It follows the proof of (Chazal et al., 2021, Theorem 9) in the i.i.d. case, with adaptations to cope with dependence using Lemma 8.

To apply Lemma 8, first note that the space $\mathcal{M}(R, M)$, endowed with the Levy-Prokhorov metric, is a Polish space (see, e.g., Prokhorov 1956, Theorem 1.11). Using Lemma 8 as in (Doukhan et al., 1995, Proof of Proposition 2) yields the existence of $\tilde{X}_1, \ldots, \tilde{X}_n$ such that, denoting by $Y_k$ (resp. $\tilde{Y}_k$) the vector $(X_{(k-1)q+1}, \ldots, X_{kq})$ (resp. $(\tilde{X}_{(k-1)q+1}, \ldots, \tilde{X}_{kq})$), for $1 \leqslant k \leqslant n/q$, it holds:

- For every $k \geqslant 1$ $Y_k$ has the same distribution as $\tilde{Y}_k$, and $\mathbb{P}(\tilde{Y}_k \neq Y_k) \leqslant \beta(q)$.
- The random variables $(Y_{2k})_{k\geqslant 1}$ are independent, as well as the variables $(Y_{2k-1})_{k\geqslant 1}$.

For any $\mathbf{c} \in \mathcal{B}(0, R)^k$, we denote by $\hat{m}(\mathbf{c})$ (resp. $\tilde{m}(\mathbf{c})$) the vector of centroids defined by, for all $j = 1, \ldots, k$,

$$\hat{m}(\mathbf{c})_j = \frac{\bar{X}_n(du)\left[u\mathbb{1}_{W_j(\mathbf{c})}(u)\right]}{\hat{p}_j(\mathbf{c})}, \qquad \tilde{m}(\mathbf{c})_j = \frac{\bar{\tilde{X}}_n(du)\left[u\mathbb{1}_{W_j(\mathbf{c})}(u)\right]}{\tilde{p}_j(\mathbf{c})},$$

where $\hat{p}_j(\mathbf{c})$ (resp. $\tilde{p}_j(\mathbf{c})$) denotes $\bar{X}_n(W_j(\mathbf{c}))$ (resp. $\bar{\tilde{X}}_n(W_j(\mathbf{c}))$), adopting the convention $\hat{m}_j(\mathbf{c}), \tilde{m}_j(\mathbf{c}) = 0$ when the corresponding cell weight is null.

The following lemma ensures that $\hat{m}(\mathbf{c})$ contracts toward $\mathbf{c}^*$, provided $\mathbf{c} \in \mathcal{B}(\mathbf{c}^*, R_0)$.

**Lemma 11** *With probability larger than $1 - 16e^{-c_1 np_{\min}/(qM)} - 2e^{-x}$, it holds, for every $\mathbf{c} \in \mathcal{B}(\mathbf{c}^*, R_0)$,*

$$\|\hat{m}(\mathbf{c}) - \mathbf{c}^*\|^2 \leqslant \frac{3}{4}\|\mathbf{c} - \mathbf{c}^*\|^2 + C_1\frac{D^2_{n/q}}{p^2_{\min}} + C_2\frac{kR^2M^2}{p^2_{\min}}\left(\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}_{X_i\neq\tilde{X}_i}\right)^2,$$

*where* $D_{n/q} = \frac{RM\sqrt{q}}{\sqrt{n}}\left(k\sqrt{d\log(k)} + \sqrt{x}\right)$.

The proof of Lemma 11 is postponed to Section 7.2.3. Equipped with Lemma 11, we first prove recursively that, if $\mathbf{c}^{(0)} \in \mathcal{B}(\mathbf{c}^*, R_0)$, then w.h.p., for all $t > 0$ $\mathbf{c}^{(t)} \in \mathcal{B}(\mathbf{c}^*, R_0)$. We let $\Omega_1$ be defined as

$$\Omega_1 = \left\{ C_2 k R^2 M^2 \left( \frac{1}{n}\sum_{i=1}^n \mathbb{1}_{X_i \neq \tilde{X}_i} \right)^2 / p_{\min}^2 \leqslant R_0^2/8 \right\}.$$

Noting that $\mathbb{E}\left(\frac{1}{n}\sum_{i=1}^n \mathbb{1}_{X_i\neq\tilde{X}_i}\right)^2 \leqslant \mathbb{E}\left(\frac{1}{n}\sum_{i=1}^n \mathbb{1}_{X_i\neq\tilde{X}_i}\right) = \beta(q)$, Markov inequality yields

$$\mathbb{P}(\Omega_1^c) \leqslant C\frac{kM^2}{\kappa_0^2 p_{\min}^2}\beta(q).$$

Choosing $x = c_1(n/q)\kappa_0^2 p_{\min}^2/M^2$ in Lemma 11, for $c_1$ small enough yields, for $(n/q)$ large enough,

$$\|\hat{m}(\mathbf{c}) - \mathbf{c}^*\|^2 \leqslant \frac{3}{4}R_0^2 + \frac{R_0^2}{8} + \frac{R_0^2}{8} = R_0^2,$$

with probability larger than $1 - 18e^{-c_1 n\kappa_0^2 p_{\min}^2/qM^2} - C\frac{kM^2}{\kappa_0^2 p_{\min}^2}\beta(q)$, provided $\mathbf{c} \in \mathcal{B}(\mathbf{c}^*, R_0)$. Denoting by $\Omega_2$ the probability event onto which the above equation holds, a straightforward recursion entails that, if $\mathbf{c}^{(0)} \in \mathcal{B}(\mathbf{c}^*, R_0)$, then, for all $t \geqslant 1$ $\mathbf{c}^{(t)} = \hat{m}(\mathbf{c}^{(t-1)}) \in \mathcal{B}(\mathbf{c}^*, R_0)$, on $\Omega_2$.

Then, using Lemma 11 iteratively yields that, on $\Omega_2 \cap \Omega_x$, where $\mathbb{P}(\Omega_x^c) \leqslant 2e^{-x}$, for all $t \geqslant 1$, provided $\mathbf{c}^{(0)} \in \mathcal{B}(\mathbf{c}^*, R_0)$,

$$\|\mathbf{c}^{(t)} - \mathbf{c}^*\|^2 \leqslant \left(\frac{3}{4}\right)^t \|\mathbf{c}^{(0)} - \mathbf{c}^*\|^2 + C_1\frac{D_{n/q}^2}{p_{\min}^2} + C_2\frac{kR^2M^2}{p_{\min}^2}\left(\frac{1}{n}\sum_{i=1}^n \mathbb{1}_{X_i\neq\tilde{X}_i}\right)^2. \tag{9}$$

Theorem 5 now easily follows. For the first inequality, let $t \geqslant 1$, then, using Markov inequality again gives

$$\mathbb{P}\left(\frac{1}{n}\sum_{i=1}^n \mathbb{1}_{X_i\neq\tilde{X}_i} \geqslant \sqrt{q/n}\right) \leqslant \sqrt{\frac{n}{q}}\beta(q).$$

Thus, the assumption $\beta^2(q)/q^3 \leqslant n^{-3}$ entails that

$$\|\mathbf{c}^{(t)} - \mathbf{c}^*\|^2 \leqslant \left(\frac{3}{4}\right)^t R_0^2 + C_1\frac{D_{n/q}^2}{p_{\min}^2} + C_2\frac{kR^2M^2}{p_{\min}^2(n/q)},$$

with probability larger than $1 - 18e^{-c_1 n\kappa_0^2 p_{\min}^2/qM^2} - C\frac{kM^2}{\kappa_0^2 p_{\min}^2}\beta(q) - q/n - 2e^{-x}$ that is larger than $1 - C\frac{qkM^2}{n\kappa_0^2 p_{\min}^2} - 2e^{-x}$.

For the second inequality, denote by $Z_t$ the random variable

$$Z_t = \left( \|\mathbf{c}^{(t)} - \mathbf{c}^*\|^2 - \left(\frac{3}{4}\right)^t \|\mathbf{c}^{(0)} - \mathbf{c}^*\|^2 - C_1 \frac{qR^2M^2k^2d\log(k)}{np_{\min}^2} - C_2 \frac{kR^2M^2}{p_{\min}^2} \left( \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{X_i \neq \tilde{X}_i} \right)^2 \right)_+ \mathbb{1}_{\Omega_2},$$

and remark that (9) entails

$$\mathbb{P}\left( Z_t \geqslant C \frac{R^2M^2q}{n} x \right) \leqslant \mathbb{P}(\Omega_x^c) \leqslant 2e^{-x}.$$

We deduce that

$$\mathbb{E}(Z_t) \leqslant C \frac{qR^2M^2}{n},$$

which leads to

$$\begin{aligned}
\mathbb{E}(\|\mathbf{c}^{(t)} - \mathbf{c}^*\|^2) &\leqslant \mathbb{E}(\|\mathbf{c}^{(t)} - \mathbf{c}^*\|^2 \mathbb{1}_{\Omega_2}) + 4kR^2M\mathbb{P}(\Omega_2^c) \\
&\leqslant \mathbb{E}(Z_t) + \left(\frac{3}{4}\right)^t \|\mathbf{c}^{(0)} - \mathbf{c}^*\|^2 + C_1 \frac{qR^2M^2k^2d\log(k)}{np_{\min}^2} \\
&\quad + C_2 \frac{kR^2M^2}{p_{\min}^2} \mathbb{E}\left( \left( \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{X_i \neq \tilde{X}_i} \right)^2 \right) + 4kR^2M\mathbb{P}(\Omega_2^c).
\end{aligned}$$

Noting that

$$\mathbb{E}\left( \left( \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{X_i \neq \tilde{X}_i} \right)^2 \right) \leqslant \mathbb{E}\left( \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{X_i \neq \tilde{X}_i} \right) = \beta(q),$$

and using

$$\begin{aligned}
\mathbb{P}(\Omega_2^c) &\leqslant \left( e^{-c_1 n \kappa_0^2 p_{\min}^2 / qM^2} + \frac{kM^2}{\kappa_0^2 p_{\min}^2} \beta(q) \right) \\
&\leqslant C \frac{kM^2}{\kappa_0^2 p_{\min}^2} (\beta(q) + (q/n)) \\
&\leqslant C \frac{qkM^2}{n\kappa_0^2 p_{\min}^2}
\end{aligned}$$

whenever $\beta(q) \leqslant q/n$ leads to the result. ■

### 7.2.2 Proof of Theorem 6

**Proof** [Proof of Theorem 6] This proof follows the steps of (Chazal et al., 2021, Proof of Lemma 18).

As in the proof of Lemma 11, let $\tilde{X}_1, \ldots, \tilde{X}_n$ be such that, denoting by $Y_k$ (resp. $\tilde{Y}_k$) the vector $(X_{(k-1)q+1}, \ldots, X_{kq})$ (resp. $(\tilde{X}_{(k-1)q+1}, \ldots, \tilde{X}_{kq})$), for $1 \leqslant k \leqslant n/q$, it holds:

- For every $k \geqslant 1$ $Y_k$ has the same distribution as $\tilde{Y}_k$, and $\mathbb{P}(\tilde{Y}_k \neq Y_k) \leqslant \beta(q)$.
- The random variables $(Y_{2k})_{k\geqslant 1}$ are independent, as well as the variables $(Y_{2k-1})_{k\geqslant 1}$.

Let $A_{\perp\!\!\!\perp}$ denote the event

$$A_{\perp\!\!\!\perp} = \left\{\forall j = 1, \dots, n/q \quad Y_j = \tilde{Y}_j\right\}.$$

A standard union bound yields that $\mathbb{P}(A_{\perp\!\!\!\perp}^c) \leqslant \frac{n}{q}\beta(q)$. On $A_{\perp\!\!\!\perp}$, the minibatches used by Algorithm 4 may be considered as independent, so that the main lines of (Chazal et al., 2021, Proof of Lemma 18) readily applies, replacing $X_i$'s by $\tilde{X}_i$'s. In what follows we let $\tilde{\mathbf{c}}^{(t)}$ denote the output of the $t$-th iteration of Algorithm 4 based on $\tilde{X}_1, \dots, \tilde{X}_n$.

Assume that $n \geqslant k$, and $q \geqslant C\frac{M^2}{p_{\min}^2}\log(n)$, for a large enough constant $C$ that only depends on $\int_0^1 \beta^{-1}(u)du$, to be fixed later. For $t \leqslant n/(4q) = T$, let $A_{t,1}$ and $A_{t,3}$ denote the events

$$A_{t,1} = \left\{\forall j = 1, \dots, k \quad |\tilde{p}_j(t) - p_j(t)| \leqslant \frac{p_{\min}}{128}\right\},$$

$$A_{t,3} = \left\{\forall j = 1, \dots, k \quad \left\|\int (\tilde{c}_j^{(t)} - u)\mathbb{1}_{W_j(\tilde{\mathbf{c}}^{(t)})}(u)(\bar{\tilde{X}}_{B_t^{(3)}} - \mathbb{E}(X))(du)\right\| \leqslant 8R\sqrt{\frac{Mkdp_{\min}}{C}}\right\},$$

where $\tilde{p}_j(t) = \bar{\tilde{X}}_{B_t^{(1)}}(W_j(\tilde{\mathbf{c}}^{(t)}))$. Then, according to Theorem 9 with $x = 2\log(n)$ and Lemma 10 to bound the corresponding $\sigma$, for $j \in \{1, 3\}$, $\mathbb{P}(A_{t,j}) \leqslant 4dk/n^2 + 2kd\beta(q_n/18)$, for $n$ large enough.

Further, define

$$A_{\leqslant t} = \bigcap_{j\leqslant t} A_{j,1} \cap A_{j,3}.$$

Then, provided that $q \geqslant c_0\frac{k^2dM^2}{p_{\min}^2\kappa_0^2}\log(n)$, where $c_0$ only depends on $\int_0^1 \beta^{-1}(u)du$, we may prove recursively that

$$\forall p \leqslant t \quad \tilde{\mathbf{c}}^{(p)} \in \mathcal{B}(\mathbf{c}^*, R_0)$$

on $A_{\leqslant t}$ whenever $\tilde{\mathbf{c}}^{(0)} = \mathbf{c}^{(0)} \in \mathcal{B}(\mathbf{c}^*, R_0)$ (first step of the proof of (Chazal et al., 2021, Lemma 18)).

Next, denoting by $a_t = \|\tilde{\mathbf{c}}^{(t)} - \mathbf{c}^*\|^2\mathbb{1}_{A_{\leqslant t}}$, we may write

$$\mathbb{E}(a_{t+1}) \leqslant \mathbb{E}(\|\tilde{\mathbf{c}}^{(t+1)} - \mathbf{c}^*\|^2\mathbb{1}_{A_{t+1,1}}\mathbb{1}_{A_{\leqslant t}}) + R_1,$$

with

$$\begin{aligned} R_1 &\leqslant 4kR^2\left(\mathbb{P}(A_{t+1,3}^c)\right) \\ &\leqslant 16k^2dR^2\left(n^{-2} + \beta(q/18)\right) \\ &\leqslant 32k^2dR^2(q/n)^2, \end{aligned}$$

recalling that $\beta(q/18)/q^2 \leqslant n^{-2}$. Proceeding as in (Chazal et al., 2021, Proof of Lemma 18), we may further bound

$$\mathbb{E}(\|\tilde{\mathbf{c}}^{(t+1)} - \mathbf{c}^*\|^2 \mathbb{1}_{A_{t+1,1}} \mathbb{1}_{A_{\leqslant t}}) \leqslant \left(1 - \frac{2 - K_1}{t+1}\right) \mathbb{E}(a_t) + \frac{12kdMR^2}{p_{\min}(t+1)^2},$$

for some $K_1 \leqslant 0.5$. Noticing that $k \leqslant M/p_{\min}$ and $t+1 \leqslant T = n/(4q)$ yields that

$$\mathbb{E}(a_{t+1}) \leqslant \left(1 - \frac{2 - K_1}{t+1}\right) \mathbb{E}(a_t) + \frac{14kdMR^2}{p_{\min}(t+1)^2}.$$

Following (Chazal et al., 2021, Proof of Theorem 10), a standard recursion entails

$$\mathbb{E}(a_t) \leqslant \frac{28kdMR^2}{p_{\min} t},$$

for $t \leqslant n/(4q)$. At last, since $\|\tilde{\mathbf{c}}^{(T)} - \mathbf{c}^*\|^2 \mathbb{1}_{A_{\perp\!\!\!\perp}} = \|\mathbf{c}^{(T)} - \mathbf{c}^*\|^2 \mathbb{1}_{A_{\perp\!\!\!\perp}}$, we conclude that

$$\begin{aligned}
\mathbb{E}\|\mathbf{c}^{(T)} - \mathbf{c}^*\|^2 &\leqslant \mathbb{E}(\|\tilde{\mathbf{c}}^{(T)} - \mathbf{c}^*\|^2) + 4kR^2 \mathbb{P}(A_{\perp\!\!\!\perp}^c) \\
&\leqslant \mathbb{E}(\|\tilde{\mathbf{c}}^{(T)} - \mathbf{c}^*\|^2 \mathbb{1}_{A_{\leqslant T}}) + 4kR^2 \mathbb{P}(A_{\leqslant T}^c) + \frac{4kR^2}{(n/q)} \\
&\leqslant \mathbb{E}(a_T) + \frac{16k^2R^2d}{(n/q)} \\
&\leqslant 128 \frac{kMR^2d}{p_{\min}(n/q)},
\end{aligned}$$

where $k \leqslant M/p_{\min}$ and $T = \frac{(n/q)}{4}$ have been used. ■

### 7.2.3 Proof of Lemma 11

**Proof** [Proof of Lemma 11] Assume that $\mathbf{c} \in \mathcal{B}(\mathbf{c}^*, R_0)$, for some optimal $\mathbf{c}^* \in \mathcal{C}_{opt}$. Then, for any $K > 0$, it holds

$$\|\hat{m}(\mathbf{c}) - \mathbf{c}^*\|^2 \leqslant (1+K)\|\tilde{m}(\mathbf{c}) - \mathbf{c}^*\|^2 + (1+K^{-1})\|\hat{m}(\mathbf{c}) - \tilde{m}(\mathbf{c})\|^2. \tag{10}$$

The first term of the right hand side may be controlled using a slight adaptation of (Chazal et al., 2021, Lemma 22).

**Lemma 12** *With probability larger than $1 - 16e^{-x}$, for all $\mathbf{c} \in \mathcal{B}(0,R)^k$ and $j \in [\![1,k]\!]$, it holds*

$$\begin{aligned}
\tilde{p}_j(\mathbf{c}) &\geqslant p_j(\mathbf{c}) - \sqrt{p_j(\mathbf{c})}\sqrt{\frac{8Mc_0q\log(k)\log(2nN_{\max})}{n} + \frac{8Mqx}{n}}, \\
\tilde{p}_j(\mathbf{c}) &\leqslant p_j(\mathbf{c}) + \frac{8Mc_0q\log(k)\log(2nN_{\max})}{n} + \frac{8Mqx}{n} \\
&\quad + \sqrt{\frac{8Mc_0q\log(k)\log(2nN_{\max})}{n} + \frac{8Mqx}{n}}\sqrt{p_j(\mathbf{c})},
\end{aligned}$$

*where $c_0$ is an absolute constant. Moreover, with probability larger than $1-2e^{-x}$, it holds*

$$\sup_{\mathbf{c}\in\mathcal{B}(0,R)^k}\left\|\left(\int (c_j-u)\mathbb{1}_{W_j(\mathbf{c})}(u)(\bar{\tilde{X}}_n-\mathbb{E}(X))(du)\right)_{j=1,\dots,k}\right\| \leqslant C_0\frac{RM\sqrt{q}}{\sqrt{n}}\left(k\sqrt{d\log(k)}+\sqrt{x}\right),$$

*where $C_0$ is an absolute constant.*

**Proof** [Proof of Lemma 12] We intend here to recover the standard i.i.d. bounds given in (Chazal et al., 2021, Lemma 22). To this aim, we let $\tilde{p}_{j,0}(\mathbf{c})$ and $\tilde{p}_{j,1}(\mathbf{c})$ be defined by

$$\tilde{p}_{j,r}(\mathbf{c})=\frac{2q}{n}\sum_{s=1}^{n/2q}\bar{\tilde{Y}}_{2s-r}(W_j(\mathbf{c})),$$

for $r\in\{0,1\}$, where $\bar{\tilde{Y}}_{2s-r}=\frac{1}{q}\sum_{t=(2s-r-1)q+1}^{(2s-r)q}\tilde{X}_t$ is a measure in $\mathcal{M}(R,M)$, with total number of support points bounded by $qN_{\max}$, and remark that

$$\tilde{p}_j(\mathbf{c})=\frac{1}{2}\left(\tilde{p}_{j,0}(\mathbf{c})+\tilde{p}_{j,1}(\mathbf{c})\right).$$

Since $\mathbb{E}(\bar{\tilde{Y}}_{2s-r})(W_j(\mathbf{c}))=p_j(\mathbf{c})$, and the $\tilde{p}_{j,r}(\mathbf{c})$'s are sums of $n/2q$ independent measures evaluated on $W_j(\mathbf{c})$, we may readily apply (Chazal et al., 2021, Lemma 22) replacing $n$ by $n/(2q)$ to each of them, leading to the deviation bounds on the $\tilde{p}_j(\mathbf{c})$'s.

For the third inequality of Lemma 12, denoting by

$$\bar{\tilde{X}}_{n,j}=\frac{2q}{n}\sum_{s=1}^{n/2q}\bar{\tilde{Y}}_{2s-j},$$

for $j\in\{0,1\}$, it holds, for any $\mathbf{c}\in\mathcal{B}(0,R)^k$,

$$\begin{aligned}\left\|\left(\int (c_j-u)\mathbb{1}_{W_j(\mathbf{c})}(u)(\bar{\tilde{X}}_n-\mathbb{E}(X))(du)\right)_{j=1,\dots,k}\right\| &\leqslant \frac{1}{2}\left(\left\|\left(\int (c_j-u)\mathbb{1}_{W_j(\mathbf{c})}(u)(\bar{\tilde{X}}_{n,0}-\mathbb{E}(X))(du)\right)_{j=1,\dots,k}\right\|\right.\\ &\left.+\left\|\left(\int (c_j-u)\mathbb{1}_{W_j(\mathbf{c})}(u)(\bar{\tilde{X}}_{n,1}-\mathbb{E}(X))(du)\right)_{j=1,\dots,k}\right\|\right).\end{aligned}$$

Since each of the $\bar{\tilde{X}}_{n,j}$'s are i.i.d. sums of discrete measures (the $\bar{\tilde{Y}}_{2s-j}$'s), (Chazal et al., 2021, Lemma 22) readily applies (with sample size $n/(2q)$), giving the result. ■

We now proceed with the first term in (10) as in (Chazal et al., 2021, Proof of Lemma 17). Using the first two inequalities of Lemma 12 with $x=c_1np_{\min}/M$ yields a probability event $\Omega_1$ onto which

$$\begin{aligned}\tilde{p}_j(\mathbf{c}) &\geqslant \frac{63}{64}p_j(\mathbf{c})-\frac{p_{\min}}{64}\geqslant\frac{31}{32}p_{\min},\\ \tilde{p}_j(\mathbf{c}) &\leqslant \frac{33}{32}p_j(\mathbf{c}^*).\end{aligned}$$

Combining this with the inequality of Lemma 12 yields, for $n$ large enough and all $\mathbf{c} \in \mathcal{B}(\mathbf{c}^*, R_0)$, with probability larger than $1 - 16e^{-c_1 n p_{\min}/n} - 2e^{-x}$,

$$\|\tilde{m}(\mathbf{c}) - \mathbf{c}^*\|^2 \leqslant 0.65\|\mathbf{c} - \mathbf{c}^*\|^2 + \frac{C}{p_{\min}^2} D_{n/q}^2. \tag{11}$$

The precise derivation of (11) may be found in (Chazal et al., 2021, Proof of Lemma 17, pp.34-35). Plugging (11) into (10) leads to, for a small enough $K$,

$$\|\hat{m}(\mathbf{c}) - \mathbf{c}^*\|^2 \leqslant \frac{3}{4}\|\mathbf{c} - \mathbf{c}^*\|^2 + \frac{C}{p_{\min}^2} D_{n/q}^2 + C_2\|\hat{m}(\mathbf{c}) - \tilde{m}(\mathbf{c})\|^2,$$

with probability larger than $1 - 16e^{-c_1 n p_{\min}/n} - 2e^{-x}$.

It remains to control the last term $\|\hat{m}(\mathbf{c}) - \tilde{m}(\mathbf{c})\|^2$. To do so, note that, for every $j = 1, \ldots, j$,

$$\begin{aligned} |\hat{p}_j(\mathbf{c}) - \tilde{p}_j(\mathbf{c})| &= \frac{1}{n}\left|\sum_{i=1}^n X_i(W_j(\mathbf{c})) - \tilde{X}_i(W_j(\mathbf{c}))\right| \\ &\leqslant \frac{M}{n}\sum_{i=1}^n \mathbb{1}_{X_i \neq \tilde{X}_i}, \end{aligned} \tag{12}$$

and

$$\left\|(\bar{X}_n(du) - \bar{\tilde{X}}_n(du))\left[u\mathbb{1}_{W_j(\mathbf{c})}(u)\right]\right\| \leqslant \frac{2RM}{n}\sum_{i=1}^n \mathbb{1}_{X_i \neq \tilde{X}_i}.$$

On $\Omega_1$, it holds, for every $j = 1, \ldots, k$,

$$\begin{aligned} \|\hat{m}_j(\mathbf{c}) - \tilde{m}_j(\mathbf{c})\| &= \left\|\frac{\bar{X}_n(du)\left[u\mathbb{1}_{W_j(\mathbf{c})}(u)\right]}{\hat{p}_j(\mathbf{c})} - \frac{\bar{\tilde{X}}_n(du)\left[u\mathbb{1}_{W_j(\mathbf{c})}(u)\right]}{\tilde{p}_j(\mathbf{c})}\right\| \\ &\leqslant \left\|\bar{X}_n(du)\left[u\mathbb{1}_{W_j(\mathbf{c})}(u)\right]\right\|\left|\frac{1}{\hat{p}_j(\mathbf{c})} - \frac{1}{\tilde{p}_j(\mathbf{c})}\right| \\ &\quad + \frac{1}{\tilde{p}_j(\mathbf{c})}\left\|(\bar{X}_n(du) - \bar{\tilde{X}}_n(du))\left[u\mathbb{1}_{W_j(\mathbf{c})}(u)\right]\right\| \\ &\leqslant R\frac{|\hat{p}_j(\mathbf{c}) - \tilde{p}_j(\mathbf{c})|}{\tilde{p}_j(\mathbf{c})} + \frac{2RM}{n\tilde{p}_j(\mathbf{c})}\sum_{i=1}^n \mathbb{1}_{X_i \neq \tilde{X}_i} \\ &\leqslant C\frac{RM}{np_{\min}}\sum_{i=1}^n \mathbb{1}_{X_i \neq \tilde{X}_i}. \end{aligned}$$

Squaring and taking the sum with respect to $j$ gives the result. ■

### 7.2.4 Proof of Proposition 7

**Proof** [Proof of Proposition 7] Let $Z_1, \dots, Z_n$ denote the sequence $\|\tilde{v}_1\|, \dots, \|\tilde{v}_n\|$, that is a stationary $\beta$-mixing sequence of real-valued random variables. For $t \in \mathbb{R}$, we let

$$F_n(t) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{Z_i > t},$$

$F(t) = \mathbb{P}(Z > t)$, and $\hat{t}$ be such that $F_n(\hat{t}) \leqslant \alpha - \delta$. In the i.i.d. case, we might bound $\sup_t (F(t) - F_n(t))/\sqrt{F_n(t)}$ using a standard inequality such as in (Boucheron et al., 2005, Section 5.1.2).

As for the proofs of Theorem 5 and 6, we compare with the i.i.d. case by introducing auxiliary variables.

We let $\tilde{Z}_1, \dots, \tilde{Z}_n$ be such that, denoting by $Y_k$ (resp. $\tilde{Y}_k$) the vector $(Z_{(k-1)q+1}, \dots, Z_{kq})$ (resp. $(\tilde{Z}_{(k-1)q+1}, \dots, \tilde{Z}_{kq})$), it holds

- For every $1 \leqslant k \leqslant n/q$ $Y_k \sim \tilde{Y}_k$, and $\mathbb{P}(Y_k \neq \tilde{Y}_k) \leqslant \beta(q)$.
- $(Y_{2k})_{k \geqslant 1}$ are independent, as well as $(Y_{2k-1})_{k \geqslant 1}$.

Let $\tilde{F}_n(t)$ denote $\sum_{i=1}^{n} \mathbb{1}_{\tilde{Z}_i > t}$. Then, for any $t \in \mathbb{R}$, we have

$$\tilde{F}_n(t) \leqslant F_n(t) + \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{Z_i \neq \tilde{Z}_i}.$$

If $\Omega_1$ is the event $\left\{ \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{Z_i \neq \tilde{Z}_i} \leqslant \sqrt{\frac{\alpha q}{n}} \right\}$, that has probability larger than $1 - \beta(q)\sqrt{n/(\alpha q)}$ (using Markov inequality as before), then on $\Omega_1$ it readily holds

$$\begin{aligned} \tilde{F}_n(\hat{t}) &\leqslant F_n(\hat{t}) + \sqrt{\frac{\alpha q}{n}} \\ &\leqslant \alpha - \delta + \sqrt{\frac{\alpha q}{n}}, \end{aligned}$$

so that we may write, on the same event,

$$\begin{aligned} F(\hat{t}) &= F(\hat{t}) - \tilde{F}_n(\hat{t}) + \tilde{F}_n(\hat{t}) \\ &\leqslant \alpha - \delta + \sqrt{\frac{\alpha q}{n}} + (F(\hat{t}) - \tilde{F}_n(\hat{t})). \end{aligned} \tag{13}$$

It remains to control the stochastic term $(F(\hat{t}) - \tilde{F}_n(\hat{t}))$. To do so, we denote by

$$\tilde{X}_{j,0}(t) = \frac{1}{q} \sum_{i=2(j-1)q+1}^{(2j-1)q} \mathbb{1}_{\tilde{Z}_i > t},$$

$$\tilde{X}_{j,1}(t) = \frac{1}{q} \sum_{i=(2j-1)q+1}^{2jq} \mathbb{1}_{\tilde{Z}_i > t},$$

for $j \in [\![1, n/(2q)]\!]$ and $t \in \mathbb{R}$. Note that, for any $\sigma \in \{0,1\}$, $\tilde{X}_{j,\sigma}$'s are i.i.d., take values in $[0,1]$, and have expectation $F(t)$. Next, we define, for $1 \leqslant j \leqslant n/2q$ and $t \in \mathbb{R}$,

$$\tilde{F}_{n,0}(t) = \frac{2q}{n} \sum_{j=1}^{n/2q} \tilde{X}_{j,0}(t),$$

$$\tilde{F}_{n,1}(t) = \frac{2q}{n} \sum_{j=1}^{n/2q} \tilde{X}_{j,1}(t),$$

and we note that $\tilde{F}_n(t) = \frac{1}{2}(\tilde{F}_{n,0}(t) + \tilde{F}_{n,1}(t))$. Since the $\tilde{F}_{n,\sigma}$'s are sums of i.i.d. random variables, the following concentration bound follows.

**Lemma 13** *For $j \in \{0,1\}$, and $x$ such that $(n/2q)x^2 \geqslant 1$, it holds*

$$\mathbb{P}\left( \sup_{t \in \mathbb{R}} \frac{(F(t) - \tilde{F}_{n,j}(t))}{\sqrt{F(t)}} \geqslant 2x \right) \leqslant 2ne^{-(n/2q)x^2}.$$

A proof of Lemma 13 is postponed to the following Section 7.2.5. Now, choosing $x = 2\sqrt{\frac{q\log(n)}{n}}$ entails that, with probability larger than $1 - 4(q/n) - \beta(q)\sqrt{n/(\alpha q)}$,

$$F(\hat{t}) \leqslant \alpha - \delta + \sqrt{\frac{\alpha q}{n}} + 4\sqrt{\frac{q\log(n)}{n}}\sqrt{F(\hat{t})},$$

which leads to

$$\sqrt{F(\hat{t})} \leqslant 2\sqrt{\frac{q\log(n)}{n}} + \sqrt{\alpha - \delta + \sqrt{\frac{\alpha q}{n}} + 4\frac{q\log(n)}{n}}.$$

Choosing $\delta \geqslant 4\sqrt{\alpha}\sqrt{\frac{q\log(n)}{n}} + \sqrt{\frac{\alpha q}{n}}$ ensures that the right-hand side is smaller than $\sqrt{\alpha}$. ■

### 7.2.5 Proof of Lemma 13

**Proof** [Proof of Lemma 13] The proof follows the one of (Chazal et al., 2021, Lemma 22) verbatim, at the exception of the capacity bound, that we discuss now. To lighten notation we assume that we have a $n/(2q)$ sample of $Y_i$'s, with $Y_i = (Z_{(i-1)q+1}, \ldots, Z_{iq}) \in \mathbb{R}^q$, and we consider the set of functionals

$$\mathcal{F} = \left\{ y = (z_1, \ldots, z_q) \mapsto \frac{1}{q} \sum_{i=1}^{q} \mathbb{1}_{z_i > t} \right\}.$$

Following (Chazal et al., 2021, Lemma 22), if $S_{\mathcal{F}}(y_1, \ldots, y_{n/(2q)})$ denotes the cardinality of $\{f(y_1), \ldots, f(y_{n/(2q)}) \mid f \in \mathcal{F}\}$, we have to bound

$$S_{\mathcal{F}}(Y_1, \ldots, Y_{n/(2q)}, Y'_1, \ldots, Y'_{n/(2q)}),$$

where the $Y_i'$'s are i.i.d. copies of the $Y_i$'s. Since, for every $y_1, \dots, y_{n/q}$, recalling that $y_i = (z_{(i-1)q+1}, \dots, z_{iq})$, it holds

$$S_{\mathcal{F}}(y_1, \dots, y_{n/q}) \leqslant \{(\mathbb{1}_{z_1 > t}, \dots, \mathbb{1}_{z_n > t}) \mid t \in \mathbb{R}\},$$

we deduce that

$$S_{\mathcal{F}}(Y_1, \dots, Y_{n/(2q)}, Y_1', \dots, Y_{n/(2q)}') \leqslant n.$$

The remaining of the proof follows verbatim (Chazal et al., 2021, Lemma 22). ■

## References


Henry Adams, Tegan Emerson, Michael Kirby, Rachel Neville, Chris Peterson, Patrick Shipman, Sofya Chepushtanova, Eric Hanson, Francis Motta, and Lori Ziegelmeier. Persistence images: a stable vector representation of persistent homology. *J. Mach. Learn. Res.*, 18:Paper No. 8, 35, 2017. ISSN 1532-4435.

Alekh Agarwal and John C. Duchi. The generalization ability of online algorithms for dependent data. *IEEE Trans. Inform. Theory*, 59(1):573–587, 2013. ISSN 0018-9448,1557-9654. doi: 10.1109/TIT.2012.2212414. URL `https://doi.org/10.1109/TIT.2012.2212414`.

Sylvain Arlot, Alain Celisse, and Zaid Harchaoui. A kernel multiple change-point algorithm via model selection. *J. Mach. Learn. Res.*, 20:Paper No. 162, 56, 2019. ISSN 1532-4435,1533-7928.

Jean-Daniel Boissonnat, Frédéric Chazal, and Mariette Yvinec. *Geometric and Topological Inference*, volume 57. Cambridge University Press, 2018.

Stéphane Boucheron, Olivier Bousquet, and Gábor Lugosi. Theory of classification: a survey of some recent advances. *ESAIM Probab. Stat.*, 9:323–375, 2005. ISSN 1292-8100,1262-3318. doi: 10.1051/ps:2005018. URL `https://doi.org/10.1051/ps:2005018`.

Anass El Yaagoubi Bourakna, Moo K. Chung, and Hernando Ombao. Modeling and simulating dependence in networks using topological data analysis, 2022.

Claire Brécheteau, Aurélie Fischer, and Clément Levrard. Robust Bregman clustering. *The Annals of Statistics*, 49(3):1679 – 1701, 2021. doi: 10.1214/20-AOS2018. URL `https://doi.org/10.1214/20-AOS2018`.

L Breiman. Random forests. 45:5–32, 2001-10. doi: 10.1023/A:1010950718922.

P. Bruillard, K. Nowak, and E. Purvine. Anomaly detection using persistent homology. In *2016 Cybersecurity Symposium (CYBERSEC)*, pages 7–12, Los Alamitos, CA, USA, apr 2016. IEEE Computer Society. doi: 10.1109/CYBERSEC.2016.009. URL `https://doi.ieeecomputersociety.org/10.1109/CYBERSEC.2016.009`.

Mathieu Carrière, Frédéric Chazal, Yuichi Ike, Théo Lacombe, Martin Royer, and Yuhei Umeda. Perslay: A neural network layer for persistence diagrams and new graph topological signatures. In *International Conference on Artificial Intelligence and Statistics*, pages 2786–2796. PMLR, 2020.

Frédéric Chazal and Vincent Divol. The density of expected persistence diagrams and its kernel based estimation. In *34th International Symposium on Computational Geometry*, volume 99 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 26, 15. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2018.

Frédéric Chazal and Bertrand Michel. An introduction to topological data analysis: fundamental and practical aspects for data scientists. *Frontiers in artificial intelligence*, 4:108, 2021.

Frédéric Chazal, Vin de Silva, and Steve Oudot. Persistence stability for geometric complexes. *Geometriae Dedicata*, 173(1):193–214, 2014.

Frédéric Chazal, Vin de Silva, Marc Glisse, and Steve Oudot. *The structure and stability of persistence modules*. Springer International Publishing, 2016.

Frédéric Chazal, Clément Levrard, and Martin Royer. Clustering of measures via mean measure quantization. *Electronic Journal of Statistics*, 15(1):2060 – 2104, 2021. doi: 10.1214/21-EJS1834. URL `https://doi.org/10.1214/21-EJS1834`.

Stéphane Chrétien, Ben Gao, Astrid Thebault-Guiochon, and Rémi Vaucher. Time topological analysis of eeg using signature theory, 2024.

Meryll Dindin, Yuhei Umeda, and Frederic Chazal. Topological data analysis for arrhythmia detection through modular neural networks, 2019.

Vincent Divol and Frédéric Chazal. The density of expected persistence diagrams and its kernel based estimation. *Journal of Computational Geometry*, 10(2):127–153, 2019.

P. Doukhan, P. Massart, and E. Rio. Invariance principles for absolutely regular empirical processes. *Ann. Inst. H. Poincaré Probab. Statist.*, 31(2):393–427, 1995. ISSN 0246-0203. URL `http://www.numdam.org/item?id=AIHPB_1995__31_2_393_0`.

Paul Doukhan. *Mixing*, volume 85 of *Lecture Notes in Statistics*. Springer-Verlag, New York, 1994. ISBN 0-387-94214-9. doi: 10.1007/978-1-4612-2642-0. URL `https://doi.org/10.1007/978-1-4612-2642-0`. Properties and examples.

Herbert Edelsbrunner and John Harer. *Computational topology: an introduction*. American Mathematical Society, 2010.

Petri G., Expert P., Turkheimer F., Carhart-Harris R., Nutt D., Hellyer P.J., and Vaccarino F. Homological scaffolds of brain functional networks. *Journal of the Royal Society Interface*, 11(101), 2014. doi: 10.1098/rsif.2014.0873.

Niklas Heim and James E. Avery. Adaptive anomaly detection in chaotic time series with a spatially aware echo state network. *ArXiv*, abs/1909.01709, 2019. URL `https://api.semanticscholar.org/CorpusID:202541761`.

Steffen Herbold. Autorank: A python package for automated ranking of classifiers. *Journal of Open Source Software*, 5(48):2173, 2020. doi: 10.21105/joss.02173. URL https://doi.org/10.21105/joss.02173.

Thi Kieu Khanh Ho, Ali Karami, and Narges Armanfard. Graph-based time-series anomaly detection: A survey. *arXiv preprint arXiv:2302.00058*, 2023.

Christoph D Hofer, Roland Kwitt, and Marc Niethammer. Learning representations of persistence barcodes. *Journal of Machine Learning Research*, 20(126):1–45, 2019.

Mia Hubert, Michiel Debruyne, and Peter J. Rousseeuw. Minimum covariance determinant and extensions. *Wiley Interdiscip. Rev. Comput. Stat.*, 10(3):e1421, 11, 2018. ISSN 1939-5108,1939-0068. doi: 10.1002/wics.1421. URL https://doi.org/10.1002/wics.1421.

Vincent Jacob, Fei Song, Arnaud Stiegler, Bijan Rad, Yanlei Diao, and Nesime Tatbul. Exathlon: a benchmark for explainable anomaly detection over time series. 14(11): 2613–2626, 2021-07. ISSN 2150-8097. doi: 10.14778/3476249.3476307. URL https://doi.org/10.14778/3476249.3476307.

Soham Jana, Jianqing Fan, and Sanjeev Kulkarni. A general theory for robust clustering via trimmed mean, 2024.

Clément Levrard. Nonasymptotic bounds for vector quantization in hilbert spaces. *Ann. Statist.*, 43(2):592–619, 04 2015. doi: 10.1214/14-AOS1293. URL https://doi.org/10.1214/14-AOS1293.

Clément Levrard. Quantization/clustering: when and why does k-means work. *Journal de la Société Française de Statistiques*, 159(1), 2018.

Shuya Li, Wenbin Song, Chao Zhao, Yifeng Zhang, Weiming Shen, Jing Hai, Jiawei Lu, and Yingshi Xie. An anomaly detection method for multiple time series based on similarity measurement and louvain algorithm. *Procedia Computer Science*, 200:1857–1866, 2022.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining*, pages 413–422, 2008. doi: 10.1109/ICDM.2008.17.

Hendrik P. Lopuhaa and Peter J. Rousseeuw. Breakdown Points of Affine Equivariant Estimators of Multivariate Location and Covariance Matrices. *The Annals of Statistics*, 19(1):229 – 248, 1991. doi: 10.1214/aos/1176347978. URL https://doi.org/10.1214/aos/1176347978.

Daniel J. McDonald, Cosma Rohilla Shalizi, and Mark Schervish. Estimating beta-mixing coefficients via histograms. *Electron. J. Stat.*, 9(2):2855–2883, 2015. ISSN 1935-7524. doi: 10.1214/15-EJS1094. URL https://doi.org/10.1214/15-EJS1094.

Ahmed Hossam Mohammed, Mercedes Cabrerizo, Alberto Pinzon, Ilker Yaylali, Prasanna Jayakar, and Malek Adjouadi. Graph neural networks in eeg spike detection. *Artificial Intelligence in Medicine*, 145:102663, 2023. ISSN 0933-3657. doi: https://doi.org/10.1016/

j.artmed.2023.102663. URL https://www.sciencedirect.com/science/article/pii/S093336572300177X.

Mehryar Mohri and Afshin Rostamizadeh. Stability bounds for stationary $\phi$-mixing and $\beta$-mixing processes. *J. Mach. Learn. Res.*, 11:789–814, 2010. ISSN 1532-4435,1533-7928.

Hernando Ombao and Marco Pinto. Spectral dependence, 2021.

John Paparrizos, Paul Boniol, Themis Palpanas, Ruey S. Tsay, Aaron Elmore, and Michael J. Franklin. Volume under the surface: a new accuracy evaluation measure for time-series anomaly detection. *Proc. VLDB Endow.*, 15(11):2774–2787, jul 2022a. ISSN 2150-8097. doi: 10.14778/3551793.3551830. URL https://doi.org/10.14778/3551793.3551830.

John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S. Tsay, Themis Palpanas, and Michael J. Franklin. Tsb-uad: an end-to-end benchmark suite for univariate time-series anomaly detection. *Proc. VLDB Endow.*, 15(8):1697–1711, apr 2022b. ISSN 2150-8097. doi: 10.14778/3529337.3529354. URL https://doi.org/10.14778/3529337.3529354.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Tuan D. Pham and Lanh T. Tran. Some mixing properties of time series models. *Stochastic Process. Appl.*, 19(2):297–303, 1985. ISSN 0304-4149,1879-209X. doi: 10.1016/0304-4149(85)90031-6. URL https://doi.org/10.1016/0304-4149(85)90031-6.

Yu. V. Prokhorov. Convergence of random processes and limit theorems in probability theory. *Teor. Veroyatnost. i Primenen.*, 1:177–238, 1956. ISSN 0040-361X.

Nalini Ravishanker and Renjie Chen. Topological data analysis (tda) for time series, 2019.

Ian W. Renner, Jane Elith, Adrian Baddeley, William Fithian, Trevor Hastie, Steven J. Phillips, Gordana Popovic, and David I. Warton. Point process models for presence-only analysis. *Methods in Ecology and Evolution*, 6(4):366–379, 2015. doi: 10.1111/2041-210X.12352. URL https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12352.

Emmanuel Rio. Covariance inequalities for strongly mixing processes. *Ann. Inst. H. Poincaré Probab. Statist.*, 29(4):587–597, 1993. ISSN 0246-0203. URL http://www.numdam.org/item?id=AIHPB_1993__29_4_587_0.

Peter J. Rousseeuw and Katrien Van Driessen. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41(3):212–223, 1999. doi: 10.1080/00401706.1999.10485670. URL https://www.tandfonline.com/doi/abs/10.1080/00401706.1999.10485670.

Martin Royer, Frederic Chazal, Clément Levrard, Yuhei Umeda, and Yuichi Ike. Atol: Measure vectorization for automatic topologically-oriented learning. In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on*

*Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 1000–1008. PMLR, 13–15 Apr 2021. URL https://proceedings.mlr.press/v130/royer21a.html.

Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. Anomaly detection in time series: A comprehensive evaluation. *Proceedings of the VLDB Endowment (PVLDB)*, 15 (9):1779–1797, 2022. doi: 10.14778/3538598.3538602.

Sondre Sørbø and Massimiliano Ruocco. Navigating the metric maze: a taxonomy of evaluation metrics for anomaly detection in time series. *Data Mining and Knowledge Discovery*, pages 1–42, 11 2023. doi: 10.1007/s10618-023-00988-8.

Cheng Tang and Claire Monteleoni. On lloyd's algorithm: New theoretical insights for clustering in practice. In Arthur Gretton and Christian C. Robert, editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 1280–1289, Cadiz, Spain, 09–11 May 2016. PMLR. URL http://proceedings.mlr.press/v51/tang16b.html.

The GUDHI Project. *GUDHI User and Reference Manual*. GUDHI Editorial Board, 2015. URL http://gudhi.gforge.inria.fr/doc/latest/.

Yuhei Umeda, Junji Kaneko, Hideyuki Kikuchi, and Dr. Kikuchi. Topological data analysis and its application to time-series data analysis. 2019. URL https://api.semanticscholar.org/CorpusID:225065707.

Phillip Wenig, Sebastian Schmidl, and Thorsten Papenbrock. Timeeval: A benchmarking toolkit for time series anomaly detection algorithms. *Proceedings of the VLDB Endowment (PVLDB)*, 15(12):3678–3681, 2022. doi: 10.14778/3554821.3554873.

Weichen Wu, Jisu Kim, and Alessandro Rinaldo. On the estimation of persistence intensity functions and linear representations of persistence diagrams. In *International Conference on Artificial Intelligence and Statistics*, pages 3610–3618. PMLR, 2024.

Takehisa Yairi, Yoshikiyo Kato, and Koichi Hori. Fault detection by mining association rules from house-keeping data. 2001.

Yu Zheng, Huan Yee Koh, Ming Jin, Lianhua Chi, Khoa T Phan, Shirui Pan, Yi-Ping Phoebe Chen, and Wei Xiang. Correlation-aware spatial–temporal graph learning for multivariate time-series anomaly detection. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.