Change Point Detection in Hadamard Spaces by Alternating Minimization

Anica Kostic
LSE

Vincent Runge
Evry Paris-Saclay University

Charles Truong
ENS Paris-Saclay

Abstract

Time series analysis of non-Euclidean data is highly challenging and crucial for many real-world applications. We address the problem of detecting multiple changes in time series within these complex data spaces. Hadamard spaces, which encompass important data spaces like positive semidefinite matrices, certain Wasserstein spaces, and hyperbolic spaces, provide the right general framework to address this complexity. We propose a computationally efficient two-step iterative optimization algorithm called HOP (Hadamard Optimal Partitioning) that detects changes in the sequence of so-called Fréchet means. Under mild conditions, the proposed method consistently estimates the change point locations. HOP is highly versatile, accommodating structural assumptions such as cyclic patterns and epidemic settings, making it unique in the literature. We validate its performance in synthetic and real-world scenarios, including applications in human gait analysis using EMG data with low SNR and behavioral analysis of animal motion.

1 INTRODUCTION

Detecting changes in time series is a complex statistical and computational challenge that has engaged scientists for decades (Chen and Gupta, 2012; Horváth and Rice, 2024; Truong et al., 2020; Wu et al., 2024). Data are often modeled by a parametric distribution from the exponential family (Lee, 1997; Cleynen and Lebarbier, 2017) allowing for strong theoretical guarantees (Gao et al., 2024; Verzelen et al., 2023; Liu et al., 2021) and efficient subquadratic algorithms (Maidstone et al., 2017; Romano et al., 2022; Kovács et al., 2023). Recently, there has been growing interest in change point analysis for time series of objects in non-Euclidean spaces. Such data types emerge in various modern applications, where traditional parametric methods are often inadequate. Some notable examples in the statistical literature include

1. Dynamic covariance matrices, which are often used to assess functional connectivity in neuroscience, particularly in the analysis of fMRI data. This approach helps to uncover temporal changes in brain connectivity patterns (Dai et al., 2019; Li and Li, 2023; Lin et al., 2019; Huang et al., 2021);

2. Time series of probability distributions, such as the distributions of stock returns (Horváth et al., 2021; Zhang et al., 2022);

3. Dynamic graphs, which model evolving networks in fields like social science, biology, and telecommunications. These include networks that represent human interactions or other phenomena based on interactions (Ranshous et al., 2015; Rossetti and Cazabet, 2018).

In each of these examples, data naturally resides in metric spaces with non-Euclidean geometry, thereby extending beyond traditional change point frameworks. These examples highlight the importance of change point analysis in diverse data spaces and the need for new methodologies capable of addressing them.

Meanwhile, statistical tools for handling random objects in general metric spaces have been steadily developed. Some notable recent approaches for non-Euclidean data include regression methods (Faraway, 2014; Schötz, 2022; Bulté and Sørensen, 2024), autoregressive time series modeling (Bulté and Sørensen, 2024), nonparametric inference (Wang et al., 2023b) and density estimation (Jeon and Van Keilegom,

Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

2023). There is limited literature on change point detection in general metric spaces. Dubey and Müller (2020) proposes a method for online detection of a single change point in metric spaces. In Jiang et al. (2024), a method for detecting change points in non-Euclidean data with temporal dependence is introduced. Additionally, Lin and Müller (2021) proposes a change point detection method using total variation regression.

In this work, we address the multiple change point problem for data in general metric spaces. Without the familiar Euclidean structures, basic concepts like the mean must be redefined, and existence or uniqueness isn't always assured. We consider Hadamard spaces, where the Fréchet mean – a generalization of the mean to non-Euclidean spaces – exists and is uniquely defined (Sturm, 2002). Our goal is to efficiently detect changes in the Fréchet mean of a sequence of random objects in Hadamard spaces. We propose a two-step iterative optimization algorithm with limited computation complexity and guaranteed termination. Under some mild conditions, we prove a consistency result for change point detection and localization and demonstrate the algorithm's favorable performance on both real and synthetic data. We explore possible noteworthy extensions, such as incorporating structural constraints, through simulations. Our method is adaptable for detecting cyclic patterns or epidemic changes, among other graph-constrained structures (see Runge et al. (2023)), setting it apart in the literature. Our algorithm, called `HOP`, is tested on real cyclic EMG data, where it successfully identifies informative patterns.

This paper is organized as follows. In Section 2, we review the main properties of Hadamard spaces and define our change point problem. In Section 3, we introduce our two-step approach based on a simplified dynamic programming algorithm. Theoretical guarantees are presented in Section 4. Section 5 provides simulation results on synthetic data and real data applications are discussed in Section 6.

2 DETECTING CHANGE POINTS WITHIN HADAMARD SPACES

2.1 Background on Hadamard Spaces

Definition 1. *A metric space $(\mathcal{H}, d)$ is called a Hadamard space if it is complete and if for each pair of points $(x_0, x_1) \in \mathcal{H}^2$ there exists a point $m \in \mathcal{H}$ such that for all points $y \in \mathcal{H}$ we have:*

$$d^2(m, y) + \frac{1}{4} d^2(x_0, x_1) \leq \frac{1}{2} d^2(y, x_0) + \frac{1}{2} d^2(y, x_1). \quad (1)$$

Hadamard spaces are also called global NPC spaces, where NPC stands for Non-Positive Curvature. To show the richness of possible applications when considering data in this space, we present some examples.

Example 1. *The space $\mathcal{S}_p$ of symmetric positive definite matrices equipped with the affine invariant metric defined as:*

$$d^2(\Sigma_1, \Sigma_2) = \| \log(\Sigma_2^{-\frac{1}{2}} \Sigma_1 \Sigma_2^{-\frac{1}{2}}) \|_F^2 = \sum_{i=1}^{p} \Big(\log(\sigma_i) \Big)^2, \quad (2)$$

for $\Sigma_1, \Sigma_2 \in \mathcal{S}_p$, where the σ_i are the (strictly positive) eigenvalues of matrix $\Sigma_2^{-\frac{1}{2}} \Sigma_1 \Sigma_2^{-\frac{1}{2}}$. See Section 6.1.9 of Bhatia (2009) for a proof that the affine-invariant metric on SPD matrices satisfies the Hadamard property.

Example 2. *A metric tree can be equipped with its geodesic distance. The distance between two nodes is the sum of the edge values that connect the two nodes (i.e. , the shortest path length). This is a Hadamard space, see (Bridson and Haefliger, 2013) Chapter 2, 1.15 (Example 5).*

Example 3. *Wasserstein space $\mathcal{W}_2(\mathbb{R})$, the space of probability distributions on the real line $\mathbb{R}$ with the 2-Wasserstein distance:*

$$d(F_1, F_2) = \left[\int_0^1 \left(F_1^{-1}(x) - F_2^{-1}(x) \right)^2 \mathrm{d}x \right]^{1/2}, \quad (3)$$

where F_1^{-1} and F_2^{-1} are the left continuous quantile functions associated with distributions F_1 and F_2. For more details see Chapter 2 in Panaretos and Zemel (2020).

Many other structures are Hadamard spaces, including hyperbolic spaces, phylogenetic trees, Hilbert spaces, and simply connected, complete Riemannian manifolds with non-positive sectional curvature. Euclidean spaces also trivially fall under this category.

The sample mean of data points $y_1, \ldots, y_T$ from a Hadamard space $(\mathcal{H}, d)$ can be defined via the distance function as $\hat{\mu} := \arg\min_\mu \sum d^2(y_i, \mu)$. While $\hat{\mu}$ may not exist or be unique in non-Euclidean metric spaces, it is well defined in Hadamard spaces and is called the sample Fréchet mean (Kendall, 1990). More generally, for a distribution p on $\mathcal{H}$, the Fréchet mean is $\mu := \arg\min_\mu \int d^2(Y, \mu) dp$. Fréchet mean of a distribution is unique if the variance of p is finite (see Sturm (2002)).

The favorable regularity properties of Hadamard spaces arise from the convexity of the distance function, which guarantees the existence of a unique mean. As a result, calculating the Fréchet mean becomes a convex optimization problem.

2.2 Model and Problem Statement

Given a Hadamard space $(\mathcal{H}, d)$, we observe a $\mathcal{H}$-valued signal $\boldsymbol{y} = (y_t)_{t=1}^T$ generated by the random vector $(Y_t)_{t=1}^T$ on $\mathcal{H}^T$, such that $\theta_t^\star := \mathbb{E}Y_t$ is piecewise constant (see Definition 3 in Appendix B for this specific expectation). Said differently, there exist $K^\star$ change point indices $t_1^\star, \dots, t_{K^\star}^\star$ (in increasing order) such that:

$$\theta_t^\star = c_k^\star \quad \text{if} \quad t_k^\star < t \leq t_{k+1}^\star \,, \, k = 0, \dots, K^\star \,, \tag{4}$$

where the $c_k^\star \in \mathcal{H}$ $(k = 0, \dots, K^\star)$ are the successive Fréchet means of $\boldsymbol{\theta}^\star = (\theta_t^\star)_{t=1}^T$ and, by convention, $t_0^\star := 0$ and $t_{K^\star+1}^\star := T$. The number of means (also called levels, states, or centers) is $\#\{c_k^*,\, k = 0, \dots, K^\star\} = \upsilon \leq K^\star + 1$. We define the estimator $\hat{\boldsymbol{\theta}} = (\hat{\theta}_t)_{t=1}^T$ based on an L_0 penalty (Maidstone et al., 2017; Fearnhead and Rigaill, 2019; Pishchagina et al., 2023) as an estimator of $\boldsymbol{\theta}^\star$ obtained by minimizing a penalized criterion:

$$\hat{\boldsymbol{\theta}} := \underset{\boldsymbol{\theta} \in \mathcal{H}^T}{\arg\min} \left[\sum_{t=1}^T d^2(y_t, \theta_t) + \beta \sum_{t=2}^T \mathbf{1}_{\{\theta_{t-1} \neq \theta_t\}} \right] , \tag{5}$$

where $\mathbf{1}$ is the indicator function and $\beta > 0$ controls the trade-off between data fidelity and the number of changes. Some other penalty functions (the second term) can be proposed, in particular with a value not proportional to the number of changes (Zhang and Siegmund, 2007; Arlot et al., 2012; Cleynen and Lebarbier, 2017; Verzelen et al., 2023). The multiscale penalty by Verzelen et al. (2023) is additive, making it easy to implement in a dynamic programming algorithm, and it promotes equispaced change points. If the number of change-points is known beforehand, the dynamic programming update rule can be modified to enforce this constraint.

The estimator is piece-wise constant and we decompose the signal into a succession of segments. The cost over each segment, say between indices i and j with state value $\hat{\theta}_t = c$, is given by $\textsc{cost}(y_i, \dots, y_j) = \left\{ \sum_{t=i}^j d^2(y_t, c) \right\}$. This value c is the segment representative value returning the smallest deviation, solution of the "standard" least squares optimization problem:

$$\textsc{cost}(y_i, \dots, y_j) = \min_{y \in \mathcal{H}} \left\{ \sum_{t=i}^j d^2(y_t, y) \right\} . \tag{6}$$

The uniqueness of the solution is ensured by the Hadamard structure. However, solving this minimization problem is a bottleneck in terms of computational time for change point detection algorithms.

2.3 Complexity Issue

Computing the cost (6) efficiently is essential for building algorithms running in tractable time complexity. There exist dynamic programming algorithms solving (5) exactly in quadratic time (Jackson et al., 2005; Killick et al., 2012). However, these methods use segment cost derived from the exponential family that can be computed in constant time, whatever segment length.

Computing the Fréchet mean for a given segment within a given Hadamard structure is particularly challenging. In Bačák (2014), a polynomial-time algorithm is proposed for some tree spaces. This time complexity makes Optimal Partitioning (Jackson et al., 2005) approach (a fundamental exact multiple change point method) time-consuming for large data. If the time for computing a segment cost of length l is of order $O(l^\alpha)$ with $\alpha \geq 1$ we get an overall complexity of order $\sum_{t=1}^T \left(\sum_{l=1}^t O(l^\alpha) \right) = O(T^{\alpha+2})$. Time complexity is then at least cubic, and potentially much more for some distances derived from Hadamard spaces.

As an example, the optimal transport problem between two histograms, as studied in Bonneel et al. (2011), was empirically shown to have a quadratic time complexity, leading to a prohibitive overall complexity of $O(T^4)$. For Symmetric Positive Definite (SPD) matrices, no closed-form solution exists for the commonly used affine-invariant and log-Euclidean metrics that respect the curved manifold structure, requiring the use of iterative methods (de Carvalho Bento et al., 2019; Fiori, 2009; Boumal, 2023).

To simplify the optimization problem (6), we propose restricting the set of possible values for the estimator $\hat{\boldsymbol{\theta}}$. This motivates the introduction of an alternating optimization algorithm, where the estimator is progressively refined with each iteration of the (fast) dynamic programming algorithm.

3 THE HOP METHOD

To avoid time-consuming Fréchet mean computations, we limit the set of possible Fréchet means to a finite subset $\mathcal{C} = \{c_1, \dots, c_V\} \subset \mathcal{H}$. If the selected state values are well-chosen, with some of them being close to the true $c_k^\star$, the solution to this restricted change point problem could closely approximate the solution to the original problem.

Definition 2. *The Hadamard estimator of our state-constrained multiple change point problem with states $\mathcal{C} = \{c_1, \dots, c_V\}$ is given by the path of state indices*

$\hat{\boldsymbol{p}}$ such that $\hat{\boldsymbol{\theta}} = \hat{\theta}_{\hat{\boldsymbol{p}}} = (c_{\hat{p}(1)}, \dots, c_{\hat{p}(T)})$ with:

$$\hat{\boldsymbol{p}} := \underset{\boldsymbol{p}\in\{1,\dots,V\}^T}{\arg\min} \left[\sum_{t=1}^{T} d^2(y_t, c_{p(t)}) + \beta \sum_{t=2}^{T} \mathbf{1}_{\{p(t-1)\neq p(t)\}} \right]. \tag{7}$$

If β vanishes, we simply fit each data y_t with its closest state value measured by metric d. This estimator can be further restricted by a graph structure as shown in the simulations presented in Section 6.

3.1 Dynamic Programming Update Rule

Given the set $\mathcal{C}$ we introduce the optimal cost for fitting partial data $y_1, \dots, y_t$ with a final state c_k at time t as the quantity $Q_k(t)$, $k = 1, \dots, V$. The cost can be updated for the next time step in constant time as follows (Maidstone et al., 2017; Runge et al., 2023).

Proposition 1. *The dynamic programming update rule for the time series $y_1, \dots, y_T$, at time $t+1$, given the last state value c_k, is given for $k \in \{1, \dots, V\}$ by:*

$$Q_k(t+1) = \min_{i=1,\dots,V} \Big\{ Q_i(t) + \beta \mathbf{1}_{\{i\neq k\}} \Big\} + d^2(y_{t+1}, c_k), \tag{8}$$

with the initial condition $Q_k(0) = 0$.

This update rule returns only the minimum cost $Q(T) = \min_k Q_k(T)$. However, what we would need is the succession of states. To that end, we also save the argument of the minimum in (8) in matrix $\tau \in \{1, \dots, V\}^{V\times T}$. For a transition from state i' to state i between time t and $t+1$, we write $\tau_i(t+1) = i'$ (row i, column $t+1$). If $i' = i$, there is no change at time t and last state c_i. Thus we can build backward the vector $\hat{\boldsymbol{p}}$ of successive state indices. With the last value $\hat{p}(T) = \arg\min_{i=1,\dots,V}\{Q_i(T)\}$ we get $\hat{p}(t) = \tau(t)_{\hat{p}(t+1)}$ where $t = T-1, \dots, 1$. The obtained vector $\hat{\boldsymbol{p}}$ is the path of states solving the penalized optimization problem (7).

This dynamic programming solution requires only $\mathcal{O}(VT)$ evaluations of d^2. Neglecting the d^2 distance complexity leads to an overall time complexity of order $O(V^2T)$ due to the V operations in the min operator (8).

3.2 Alternating Optimization and `HOP` Algorithm

One does not expect to get a relevant fit by using an arbitrary vector of state values $\mathcal{C}$, even if some wise choice may be made at the initial step (using the k-means algorithm, for example). We improve such choice in an iterative manner by using a two step algorithm similar to k-means or the EM algorithms. We search for the best path $\boldsymbol{p}$ (associating each data point to a center) and then refine the centers $\boldsymbol{c}$ by computing new Fréchet means. In the initial step, $\boldsymbol{c}^{(0)}$ is known (chosen in some manner) and $\boldsymbol{p}^{(0)}$ is unknown. We define:

$$\Phi : \begin{cases} \mathcal{H}^V \times \{1, \dots, V\}^T \longrightarrow \mathbb{R} \\ (\boldsymbol{c}, \boldsymbol{p}) \longmapsto \displaystyle\sum_{t=1}^{T} d^2(y_t, c_{p(t)}) + \beta \sum_{t=2}^{T} \mathbf{1}_{\{p(t-1)\neq p(t)\}}. \end{cases} \tag{9}$$

The sequence of alternating problems for inferring the path and state values is such that, for $i \in \mathbb{N}$:

$$\begin{cases} \boldsymbol{p}^{(i+1)} := \underset{\boldsymbol{p}\in\{1,\dots,V\}^T}{\arg\min}\, \Phi(\boldsymbol{c}^{(i)}, \boldsymbol{p}), \\ \boldsymbol{c}^{(i+1)} := \underset{\boldsymbol{c}\in\mathcal{H}^V}{\arg\min}\, \Phi(\boldsymbol{c}, \boldsymbol{p}^{(i+1)}). \end{cases} \tag{10}$$

The update $\boldsymbol{p}^{(i+1)}$ is computed using the dynamic programming algorithm (Section 3.1). The update $\boldsymbol{c}^{(i+1)}$ is obtained by finding the Fréchet mean state by state, that is, for the j-th state, by solving:

$$c_j^{(i+1)} = \underset{\mu\in\mathcal{H}}{\arg\min} \sum_{t=1}^{T} \mathbf{1}_{\{\boldsymbol{p}_t^{(i+1)} = j\}} d^2(\mu, y_t).$$

If the sequences $\boldsymbol{p}^{(i)}$ and $\boldsymbol{p}^{(i+1)}$ are the same, then the algorithm has converged to a (local) minimizer.

Our algorithm converges in a finite number of steps, as proven in Theorem 1. This is due to the finite number of possible breakpoint configurations and the fact that the cost function strictly decreases at each iteration, forbidding us to return to previously visited paths.

We refer to this alternating method as the `HOP` algorithm, which stands for Hadamard Optimal Partitioning. Its full description is presented in Algorithm 1, where the number of iterations κ is fixed. A version with an additional stopping criterion, such as $d^2(\boldsymbol{c}^{(i)}, \boldsymbol{c}^{(i-1)}) \leq \epsilon$, can be easily proposed.

3.3 Link to HMMs and Extensions

The proposed alternating optimization method is closely related to the method in (Bemporad et al., 2018) for fitting piecewise constant models to Euclidean data. By fixing the state values $\mathcal{C}$ we implicitly assume the existence of hidden states, thus allowing for an interpretation of (5) in terms of a Hidden Markov Model (HMM), as detailed in Section E.3 of the Appendix.

The algorithm resulting from (8) is a Viterbi algorithm in the sense that it identifies the best-fitting sequence of states for a specific HMM. The alternating optimization problem (10) shares some similarities with the Baum-Welch EM algorithm used for estimating

Algorithm 1 HOP algorithm (in Hadamard space $\mathcal{H}$)

Require: Time series $(y_i)_{i=1}^T$, penalty value $\beta > 0$

Fixed number of iterations κ
Set of initial state values $\{c_1^0, \dots, c_V^0\} \subset \mathcal{H}$
for $i = 1, \dots, \kappa$ **do**
  $Q_0 \leftarrow 0^T, \quad \mathcal{T}_1 \leftarrow \{0\}$
  for $t = 1, \dots, T$ **do**
    for $k = 1, \dots, V$ **do**
      $Q_k(t) = \min_{i=1,\dots,V} \left\{ Q_i(t-1) + \beta \mathbf{1}_{\{i \neq k\}} \right\} + d^2(y_t, c_k)$
      $\hat{\tau}_k(t) = \arg\min_{i=1,\dots,V} \left\{ Q_i(t-1) + \beta \mathbf{1}_{\{i \neq k\}} \right\}$
    end for
  end for
  Backtracking the path. We get in state j:
  $I_j \subset \{1 \dots, T\}$ indices, then $\{c_1^i, \dots, c_V^i\} =$
  $\left(\arg\min_{y \in \mathcal{H}} \sum_{t \in I_1} d^2(y_t, y), \dots, \arg\min_{y \in \mathcal{H}} \sum_{t \in I_V} d^2(y_t, y) \right)$
end for
return the last obtained path and the states $\{c_1^\kappa, \dots, c_V^\kappa\}$

HMM parameters: (E-step) optimizing the cost function (finding $\boldsymbol{p}$), and (M-step) computing new state values given the change points (finding $\boldsymbol{c}$).

However, the classical use in the change point community has some conceptual differences. (i) Rather than maximizing the likelihood and updating probabilities, our method minimizes the cost function Φ, thereby shifting the interpretation to a (more natural) segmentation cost. (ii) Transition probabilities are not estimated but are set by the penalty term β. (iii) Many transition patterns, constraining the available sequence of states, can be used, such as incorporating cyclic and epidemic structures in the sequence of Fréchet means.

Practically, constraints can be enforced through the penalty term in (5) by assigning an infinite value to specific state transitions costs, which can be easily implemented into the recurrence relation (8) in the HOP algorithm. Such constrained transitions are analogous to constrained HMM models (Roweis, 1999) or graph-constrained change point detection (Runge et al., 2023).

4 STATISTICAL ANALYSIS

4.1 Termination of the HOP Algorithm

We define the optimal value at each iteration as $\Phi^{(i)} := \Phi(\boldsymbol{c}^{(i)}, \boldsymbol{p}^{(i)})$. If the stopping criterion is $d^2(\boldsymbol{c}^{(i+1)}, \boldsymbol{c}^{(i)}) = \sum_{j=1}^V d^2(c_j^{(i+1)}, c_j^{(i)}) \leq \epsilon$ we obtain the following result.

Theorem 1. *Our algorithm terminates in at most $\kappa = \lceil (\Phi^{(1)} - \Phi^{(\infty)})/\epsilon \rceil$ iterations, where $\Phi^{(\infty)}$ is the finite limit of $(\Phi^{(i)})$ for large i. That is $d^2(\boldsymbol{c}^{(\kappa+1)}, \boldsymbol{c}^{(\kappa)}) \leq \epsilon$.*

4.2 Change Point Consistency

We define a fixed-state function: $\phi(\boldsymbol{p}) = \Phi(\boldsymbol{c}, \boldsymbol{p})$. The optimal path $\hat{\boldsymbol{p}}$ is defined by $\hat{\boldsymbol{p}} := \arg\min_{\boldsymbol{p} \in \{1,\dots,V\}^T} \phi(\boldsymbol{p})$. Recall that $\hat{\boldsymbol{\theta}} := (c_{\hat{p}(t)})_{t=1}^T$ is the Hadamard estimator defined in (7). Using the following assumptions, we establish consistency results for both the number of change-points and their localization.

Assumption 1. *(model structure) There exist*

- *a constant $\delta_{\min} \in (0, 1)$ such that $(t_{k+1}^\star - t_k^\star)/T > \delta_{\min}$ for all $k = 0, \dots, K^\star$ (minimum spacing between changes),*
- *a constant $\Delta_{\min} > 0$ such that $d^2(c_k^\star, c_{k+1}^\star) \geq \Delta_{\min}$ for all $k = 0, \dots, K^\star - 1$ (minimum jump size).*

Assumption 2. *(noise structure) Define $\varepsilon_t(\theta) := d^2(Y_t, \theta) - \mathbb{E}\left[d^2(Y_t, \theta)\right]$. For any (t, t') with $t \neq t'$, and any (θ, θ') in $\mathcal{H}^2$, the random variables $\varepsilon_t(\theta)$ and $\varepsilon_{t'}(\theta')$ are independent. Their variances are finite and bounded above uniformly over $t \in \{1, \dots, T\}$ and $\theta \in \mathcal{C}$, i.e.*

$$\max_{t=1,\dots,T} \max_{\theta \in \mathcal{C}} \mathbb{E}\left[\varepsilon_t(\theta)^2\right] < \sigma^2/4 < \infty\,, \tag{11}$$

for a numerical constant $\sigma^2 \geq 0$ depending on the data distribution.

This setting is more general than Lin and Müller (2021), where that data are assumed to be independent (we require independence through the distance function only) and the deviation from the true signal is assumed to be uniformly sub-Gaussian (we assume that ε's are square integrable).

For the next assumption we define $P(c_k^\star) = \arg\min_{c \in \mathcal{C}} \{d^2(c, c_k^\star)\}$. That is, for each true state $c_k^\star$, $P(c_k^\star) \in \mathcal{C}$ is its corresponding closest available state.

Assumption 3. *(state structure) For any pair of times (t, t') in successive segments such that $\mathbb{E}[Y_t] = c_k^\star$ and $\mathbb{E}[Y_{t'}] = c_{k+1}^\star$ with $c_k = P(c_k^\star)$ and $c_{k+1} = P(c_{k+1}^\star)$, we assume that:*

$$d^2(c_k, c_k^\star) + d^2(c_{k+1}, c_{k+1}^\star) \leq \frac{1}{4} d^2(c_k^\star, c_{k+1}^\star) \tag{12}$$

and with $\gamma \in [0, \frac{\Delta_{\min}\delta_{\min}}{12})$,

$$d^2(c_k^\star, c_k) + \gamma \geq \mathbb{E}\left[d^2(Y_t, c_k) - d^2(Y_t, c_k^\star)\right] \geq d^2(c_k^\star, c_k)\,. \tag{13}$$

Equation (12) imposes a closeness constraint between the consecutive projection states and the consecutive true states. Furthermore, this assumption ensures that the closest projection path has the same number and locations of change-points as the true one by not allowing for two consecutive states to map to the same projection state $P(c_k^\star) \neq P(c_{k+1}^\star)$. In Hilbert spaces, Equation (13) holds with $\gamma = 0$, resulting in equalities on both sides. However, in general Hadamard spaces, only the right-hand side (RHS) of (13), known as the variance inequality, always holds. The left-hand side (LHS), resembling a reverse variance inequality (see Proposition (2) in Appendix B), imposes constraints on several combined factors: the curvature of the space, the fourth moment of the distributions of Y_t, and the distance between the true levels and the projections. All these factors must remain of small order.

Define $H(\mathcal{A}, \mathcal{B})$ as the Hausdorff distance between two sets $\mathcal{A} \subset \mathbb{R}$ and $\mathcal{B} \subset \mathbb{R}$:

$$H(\mathcal{A}, \mathcal{B}) = \max \left\{ \sup_{b \in \mathcal{B}} \inf_{a \in \mathcal{A}} |a - b|, \sup_{a \in \mathcal{A}} \inf_{b \in \mathcal{B}} |a - b| \right\}. \tag{14}$$

We obtain the following convergence theorem.

Theorem 2. *Assume that Model 4, Assumption 1, Assumption 2 and Assumption 3 hold. Let $\hat{t}_k$, for $k = 1, \dots, \hat{K}$ be the change points defined by (7) where $\beta = \lambda T$ and λ is a constant such that $0 < \lambda < \frac{1}{K^\star}(\frac{\Delta_{\min}\delta_{\min}}{12} - \gamma)$. Then, we have:*

$$\lim_{T \to \infty} \mathbb{P}(\hat{K} = K^\star) = 1\,, \tag{15}$$

and for any $\delta \in (0, 1)$,

$$\mathbb{P}\left(H(\{t_k^\star\}, \{\hat{t}_k\})/T \geq \delta\right) \leq \sigma^2 \frac{C_H}{T}\left(1 + \frac{C'_H}{\delta^2}\right), \tag{16}$$

for two positive constants C_H and C'_H.

Proofs of Theorem 1 and Theorem 2 are provided in Appendix B.

5 SIMULATION STUDY

This section presents `HOP`'s performance on synthetic data. Additional details and simulations can be found in Appendix E.

5.1 $\mathcal{S}_8$ space of SPD matrices

Data. Observations belong to $\mathcal{S}_8$, the space of SPD 8×8 matrices; they follow a Wishart distribution with 10 degrees of freedom. Each time series has length $T \in \{10^3, 2 \times 10^3, 3 \times 10^3, 4 \times 10^3, 5 \times 10^3, 6 \times 10^3, 7 \times 10^3, 8 \times 10^3, 9 \times 10^3, 10^4\}$ and $K^\star$ change-points with $K^\star \in \{2, 4, 6, 8\}$. For each combination of $(T, K^\star)$, we generate 25 signals. The change point positions are randomly drawn from a Dirichlet distribution. On each segment, the observations follow a Wishart distribution with a fixed scale matrix. The mean is random and equal to $\boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^\top$ where $\boldsymbol{D}$ is a diagonal matrix whose diagonal elements follow a uniform distribution on $[0.1, 10]$ and $\boldsymbol{U}$ is an orthogonal matrix which follows the Haar distribution (uniform distribution on the orthogonal group). In total, there are 1000 time series.

Baselines. We compare `HOP` with two methods: the first, `OCP-Rie`, presented in (Wang et al., 2023a, 2024), is an Online Change Point detection algorithm for Riemannian manifolds that we adapt to the offline setting. The second method, `gSeg` Chen and Zhang (2015), is designed to detect changepoints in general metric spaces by only considering the matrix of pairwise distances between observations. Implementation for the "at most one change" setting is available online; we extend it to the multiple change point setting using a binary segmentation strategy Olshen et al. (2004), as recommended by the authors. For all three algorithms, we assume that the number of changes is known a priori. The number of iterations of `HOP` is set to $\kappa = 5$, and the metric is the affine-invariant distance (2).

Metrics. A true positive is a well-detected change point. Formally, a change point index $t^\star$ is a true positive if there is an estimated change point $\hat{t}$ such that $|t^\star - \hat{t}|$ is below a position margin. (We add the constraint that an estimated atom can only detect one true atom.) We set the position margin to 1% of the signal length. We can compute precision/recall and the F1 score from the number of true positives.

Results. As shown on Figure 1, `HOP` and `gSeg` have perfect scores, contrary to `OCP-Rie`, whose performance decreases sharply with shorter signals. As for the execution time, `HOP` and `OCP-Rie` have linear complexity, with `HOP` being sensibly faster. `gSeg` is much slower because of its quadratic complexity (in time and space). For instance, for 10^4 observations, `HOP` takes 9 seconds on average, `OCP-Rie` takes 17 seconds, and `gSeg` takes 1h35min.

5.2 $\mathcal{W}_2(\mathbb{R})$ Wasserstein space

Data. Observations belong to $\mathcal{W}_2(\mathbb{R})$, the space of probability distributions on $\mathbb{R}$, equipped with the 2-Wasserstein distance. In practice, each observation is a histogram with 30 bins generated by sampling 1000 times from a variable $\mathcal{N}(\mu, 1)$ with μ varying from segment to segment. The shift from one segment to the next is uniformly drawn from $[0.01, 0.1]$. As before,

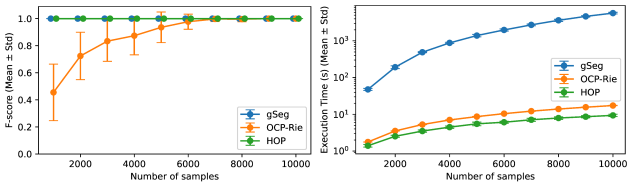


Figure 1: Results on the synthetic data set: F-score (left) and execution time (right) with respect to the number of samples.

each time series has length $T \in \{10^3, 2\times10^3, 3\times10^3, 4\times 10^3, 5\times10^3, 6\times10^3, 7\times10^3, 8\times10^3, 9\times10^3, 10^4\}$ and $K^\star$ change-points with $K^\star \in \{2, 4, 6, 8\}$. For each combination of $(T, K^\star)$, we generate 25 signals. The change point positions are randomly drawn from a Dirichlet distribution.

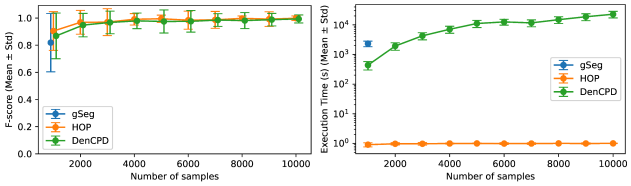


Figure 2: Results on the synthetic data set: F-score (top) and execution time (bottom) with respect to the number of samples.

Baselines. In addition to `gSeg` we add a new baseline to our simulation study: `DenCPD` as proposed in Dubey and Müller (2020). This method can detect mean and variance changes in a Hadamard space. Implementation for the "at most one change" setting is available online; we extend it to the "multiple change point" setting using a binary segmentation strategy Olshen et al. (2004).

Results. Note that there is only one point for `gSeg` in Figure 2 due to its slow runtime. For instance, processing 2000 observations takes well over 2 hours. As shown in Figure 2, all methods have similar performance in terms of F-score. We also observe that signals with more samples are easier to segment. However, our approach is significantly faster to execute: for 1000 observations, `HOP` takes less than a second, `DenCPD` takes around 7 minutes, and `gSeg` around 38 minutes.

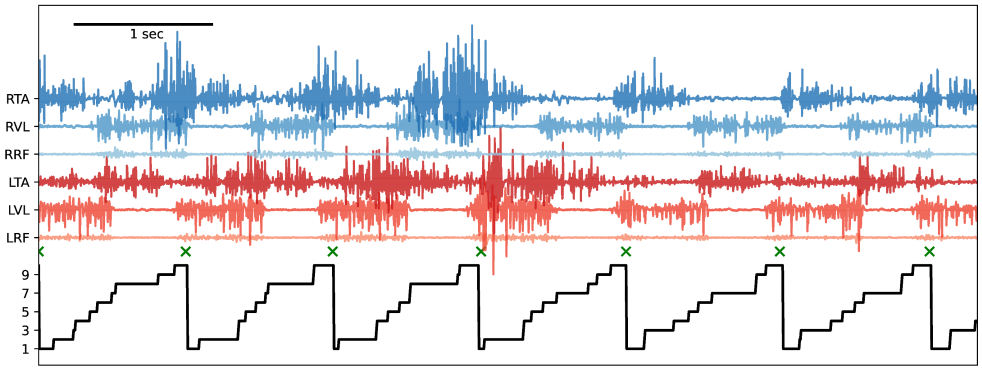
6 NEUROSCIENCE APPLICATION

6.1 Human Gait Analysis With EMG Signals

Electromyography (EMG) is a common tool to study the connections between muscle activity patterns and complex movements, such as locomotion. EMG measures the electrical muscle response to a nerve's stimulation. Researchers use it to better characterize several physiological phenomena, like muscle fatigue or adaptation strategies from subjects with neurological impairments (Rosa et al., 2014; Marco et al., 2017; Rampichini et al., 2020). (Veer and Agarwal, 2015). Our aim in this application is to decode long EMG time series collected on subjects who walk in different manners. This task is challenging because EMG signals are noisy and often contain artifacts (Boyer et al., 2023). A number of works in the literature focus on classifying small segments of EMG time series into prototypical movements such as lifting a hand or squeezing (Asanza et al., 2018; Khan et al., 2019; Ijaz and Choi, 2018). Our setting is more complex because locomotion signals are sequences of prototypical movements whose durations can vary during the exercise. To help with the decoding task, our algorithm leverages the cyclic nature of human locomotion, wherein the same sequence of movements repeats throughout the walking process.

Data and calibration. We use surface EMG data described in (Scherpereel et al., 2023). Those signals were collected on 11 subjects who walked with different styles (backward walk, heel walking, walking on an inclined floor, normal walk, toe walking, walking butt kicks, and walking with high knees) and at different speeds. Signals are sampled at 2000 Hz by sensors placed on each leg. They have 16 dimensions (8 sensors per leg), and all last around 20 seconds. A common procedure is to look for changes in the correlation structure of EMG signals (Cabrieto et al., 2018). We compute a sliding covariance (16×16 matrix) on segments of 80 ms (160 samples) and downsample to 200 Hz. This yields 180 $\mathcal{S}_{16}$-valued signals containing around 4000 observations, on which we perform change point detection with `HOP`. The authors of (Scherpereel et al., 2023) provide signals from other sensors (namely, pressure insoles) that allow us to find the start and end of each footstep. This is our ground truth to assess `HOP`'s performance. As for calibration, we set the number of states to $V = 10$ and the number of iterations to $\kappa = 5$; for each signal, the states are initialized with the first footstep (we use the labels for the first footstep only). We restrict the transitions to be cyclic (see Section 3.3), meaning that we only allow the following sequence of states: $1 \to 2 \to \cdots \to 10 \to 1 \to 2 \to \cdots \to 10 \to 1 \to 2 \ldots$ The penalty is set to $\beta = 0$.

Discussion. We only comment on `HOP`'s performance because, to the best of the authors' knowledge, there is no change point detection algorithm that can



(a)

Precision	$94.2 \pm 1.5\%$
Recall	$95.2 \pm 1.3\%$
F-score	$94.7 \pm 1.4\%$

(b)

Figure 3: (a) EMG signal example (8 seconds). Only 6 dimensions out of 16 are shown from sensors placed on the tibialis anterior (TA), vastus lateralis (VL), and the rectus femoris (RF) on the left leg ("L", in red), and the right leg ("R", in blue). Here, the subject walks backward; the green crosses delimit the true steps. The black curve shows the sequence of states learned by our method. The true steps coincide with the change from State 10 to State 1. (b) Performance of `HOP` on the EMG data set (margin 75 ms).

integrate the cyclic a priori and work on $\mathcal{S}_{16}$-valued signals. As shown on Figure 3-b, the F-score on the whole data set is 94.7%, meaning that most footsteps are recovered by our method. Some walking patterns are more difficult than others: namely, heel and toe walking are more difficult (see scores per walking type in Section E.5). To appreciate the difficulty of the task, a signal example is shown on Figure 3-a, along with the learned state sequence. Visually, there are bursts occurring regularly, as a consequence of the repeated footsteps. However, the cyclic nature of this multidimensional signal is not obvious in this representation. In this particular example, going into the space of covariance matrices and enforcing cyclic transitions allows us to perfectly detect each footstep. By examining the state sequences, we can observe that the footsteps are not identical; they vary in the amount of time spent in each state.

6.2 Behavioral Analysis of Animal Motion

Behavioral neuroscience studies the link between the brain and behavior. In the literature, a common approach consists in monitoring rodents with cameras and compressing the obtained video into a sequence of fine-grained actions, called action syllables (Wiltschko et al., 2020; Luxem et al., 2022; Shi et al., 2021). Those syllables include actions like "turning to the right", "standing still", "rearing" and can be very short (subsecond to second timescale) (Weinreb et al., 2024b). Those methods usually extract the rodent's body position (head, tail, spine, legs, etc.) with deep-learning-based computer vision algorithms and classify the body dynamics into several action classes. The output is a sequence of action syllables (1 video frame=1 syllable) and is often called an ethogram in this context. For a 30-minute video, this yields an ethogram with around 50000 syllables, which remains complex to analyze. Recently, researchers have argued that ethograms should be studied at different timescales, and they propose to organize action syllables into a hierarchical structure (Kaplan et al., 2020; Luxem et al., 2022). Intuitively, they create a tree whose leaves are the action syllables; the parents are aggregations of syllables, which we call meta-syllables. Meta-syllables encode higher-level behaviors, e.g. , stationery, grooming, exploration (Luxem et al., 2022). Nodes close to the tree root are associated with complex behaviors. We propose to use `HOP` to filter long and noisy ethograms; our objective is to limit the number of activity transitions, to yield an intuitive temporal summary of a rodent's behavior. Our approach is able to take into account the hierarchical relation between syllables.

Data. We use a video of a mouse filmed in a cage for almost 30 minutes. The monitoring protocol is described in Weinreb et al. (2024b,a). Videos are processed by DeepLabCut (Wiltschko et al., 2020) to extract eight 2D marker positions of the mouse's limbs. Time series of marker positions are decomposed using Keypoint-Moseq (Weinreb et al., 2024b): each frame is classified into a behavioral syllable. We use the procedure of Luxem et al. (2022) to organize syllables into a tree. Intuitively, two syllables with many transitions from one another are merged to create a "meta-syllable". This procedure is iterated until only a single meta-syllable remains (the tree root). The leaves of this tree are "true" syllables, meaning that they can be found in the ethogram; other nodes of this tree are meta-syllables in the sense that they are an aggrega-

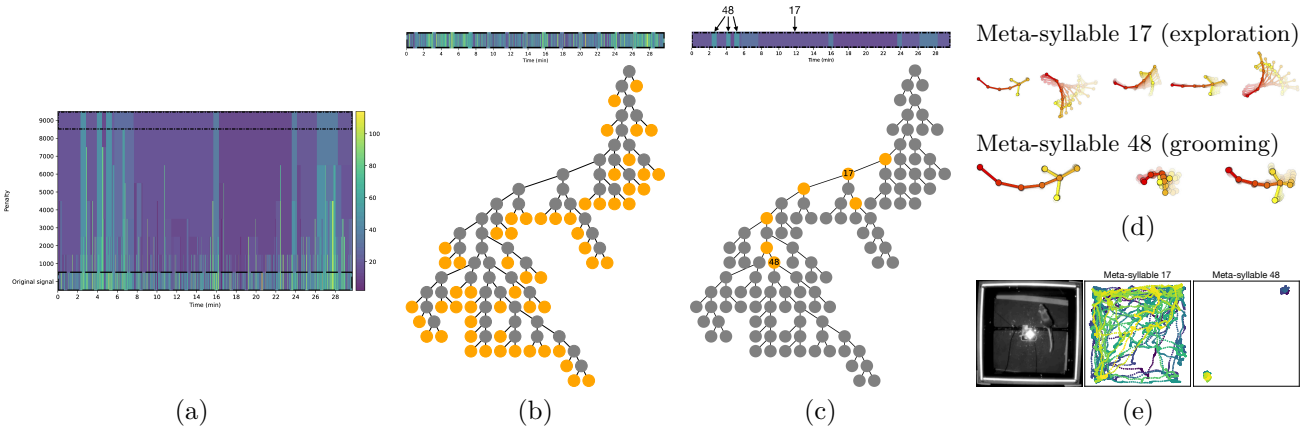


Figure 4: (a) Filtered ethograms for several penalty values. The original (unfiltered) ethogram is at the bottom; the most filtered is at the top. Each color is an action syllable. (b) The original ethogram is shown, and below is the hierarchical representation of action syllables. Orange nodes correspond to syllables that are present in the original ethogram. By construction, they correspond to leaves. (c) The most filtered ethogram is shown, and below, the hierarchical representation of action syllables. Orange nodes correspond to syllables that are present in the most filtered ethogram. The orange nodes are closer to the root node, meaning that they encode higher-level semantics. Two particular (meta-)syllables, 17 and 48, are highlighted in the ethogram and the tree. (d) Most frequent action syllables associated with Meta-syllables 17 and 48, which respectively correspond to "exploration" and 'grooming". For each action syllable, the average 8-point mouse skeleton is shown. (e) Image of the mouse and its cage, and scatter plot of mouse's center of mass while in Meta-syllables 17 and 48. Trajectories here are color-coded: the starting (resp. end) point is depicted in blue (resp. yellow). This representation confirms that Meta-syllable 17 corresponds to exploration (the mouse went over the whole cage) and Meta-syllable 48 to grooming (the mouse stayed in the corners).

tion of "true" syllables. The computed tree defines a Hadamard space, where the distance between nodes is the shortest path length.

Discussion. As before, we only discuss HOP's result because, to the best of the authors' knowledge, there is no change point detection algorithm that can cope with observations on a tree. We apply HOP with several penalty values on the ethogram. The results are shown on Figure 4. In a nutshell, we are able to compress the original ethogram, which contains 50 distinct syllables (Figure 4-b), into a high-level ethogram with only 7 distinct meta-syllables (Figure 4-c). This filtered ethogram is easier to understand in one glimpse. We highlight two of the meta-syllables found and show that they are interpretable, as they are related to two well-known behaviors: exploration and grooming.
In order to explore finer time scales, one would only need to adjust the penalty value, to have a filtered ethogram closer to the original one.

7 CONCLUSION

In this paper, we have introduced a change point detection algorithm that can cope with the geometry of non-Euclidean signals. Our method, HOP, is able to detect location changes for time series with observations in Hadamard spaces. Our implementation is efficient, with linear complexity and theoretically sound, under mild assumptions.

We show through two challenging applications in neuroscience that our approach can reveal complex temporal structures in time series. In noisy high-dimensional EMG signals, we can extract, with high accuracy, repeating patterns of variable length; each pattern is the correlation signature of footsteps. We are also able to summarize in an interpretable way an ethogram extracted from a mouse monitoring video by embedding action syllables in a tree. Our methodology is versatile and can encode domain knowledge to yield high-level time series representation.

References

Sylvain Arlot, Alain Celisse, and Zaïd Harchaoui. A kernel multiple change-point algorithm via model selection. *J. Mach. Learn. Res.*, 20:162:1–162:56, 2012.

Víctor Asanza, Enrique Peláez, Francis Loayza, Iker Mesa, Javier Díaz, and Edwin Valarezo. Emg signal processing with clustering algorithms for motor gesture tasks. In *2018 IEEE Third Ecuador Technical Chapters Meeting (ETCM)*, pages 1–6, 2018.

M. Bačák. Computing medians and means in Hadamard spaces. *SIAM Journal on Optimization*, 24(3), 2014.

A. Bemporad, V. Breschi, D. Piga, and S. P. Boyd. Fitting jump models. *Automatica*, 96:11–21, 2018.

Rajendra Bhatia. *Positive Definite Matrices*. Princeton University Press, 2009.

Nicolas Bonneel, Michiel Van De Panne, Sylvain Paris, and Wolfgang Heidrich. Displacement interpolation using lagrangian mass transport. In *Proceedings of the 2011 SIGGRAPH Asia conference*, pages 1–12, 2011.

Nicolas Boumal. *An introduction to optimization on smooth manifolds*. Cambridge University Press, 2023.

Marianne Boyer, Laurent Bouyer, Jean-Sébastien Roy, and Alexandre Campeau-Lecours. Reducing noise, artifacts and interference in single-channel emg signals: A review. *Sensors*, 23(6):2927, 2023.

Martin R Bridson and André Haefliger. *Metric spaces of non-positive curvature*, volume 319. Springer Science & Business Media, 2013.

Matthieu Bulté and Helle Sørensen. Medoid splits for efficient random forests in metric spaces. *Computational Statistics & Data Analysis*, 198:107995, 2024.

Matthieu Bulté and Helle Sørensen. An autoregressive model for time series of random objects. *arXiv preprint arXiv:2405.03778*, 2024.

Jedelyn Cabrieto, Francis Tuerlinckx, Peter Kuppens, Frank H Wilhelm, Michael Liedlgruber, and Eva Ceulemans. Capturing correlation changes by applying kernel change point detection on the running correlations. *Information Sciences*, 447:117–139, 2018.

H. Chen and N. Zhang. Graph-based change-point detection. *The Annals of Statistics*, 43, 2 2015.

Jie Chen and Arjun K Gupta. *Parametric Statistical Change Point Analysis: With Applications to Genetics, Medicine, and Finance*. Birkhäuser Boston, 2012.

Alice Cleynen and Emilie Lebarbier. Model selection for the segmentation of multiparameter exponential family distributions. *Electronic Journal of Statistics*, 11(1):800–842, 2017.

Mengyu Dai, Zhengwu Zhang, and Anuj Srivastava. Discovering common change-point patterns in functional connectivity across subjects. *Medical Image Analysis*, 58:101532, 2019.

Glaydston de Carvalho Bento, Sandro Dimy Barbosa Bitar, João Xavier da Cruz Neto, Paulo Roberto Oliveira, and João Carlos de Oliveira Souza. Computing riemannian center of mass on hadamard manifolds. *Journal of Optimization Theory and Applications*, 183(3):977–992, 2019.

Paromita Dubey and Hans-Georg Müller. Fréchet change-point detection. *The Annals of Statistics*, 48(6):3312–3335, 2020.

Julian J Faraway. Regression for non-Euclidean data using distance matrices. *Journal of Applied Statistics*, 41(11):2342–2357, 2014.

Paul Fearnhead and Guillem Rigaill. Changepoint detection in the presence of outliers. *Journal of the American Statistical Association*, 114(525):169–183, 2019.

Simone Fiori. Learning the fréchet mean over the manifold of symmetric positive-definite matrices. *Cognitive computation*, 1:279–291, 2009.

Zhanzhongyu Gao, Xun Xiao, Yi-Ping Fang, Jing Rao, and Huadong Mo. A selective review on information criteria in multiple change point detection. *Entropy*, 26(1):50, 2024.

Lajos Horváth and Gregory Rice. *Change Point Analysis for Time Series*. Springer, 2024.

Lajos Horváth, Piotr Kokoszka, and Shixuan Wang. Monitoring for a change point in a sequence of distributions. *The Annals of Statistics*, 49(4):2271–2291, aug 2021.

Zhuobin Huang, Tingting Dan, Yi Lin, Jiazhou Chen, Hongmin Cai, and Guorong Wu. Detecting brain state changes via manifold mean shifting. In *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1335–1338, 2021.

A Ijaz and J Choi. Anomaly Detection of Electromyographic Signals. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(4):770–779, 2018.

Brad Jackson, Jeffrey D Scargle, David Barnes, Sundararajan Arabhi, Alina Alt, Peter Gioumousis, Elyus Gwin, Paungkaew Sangtrakulcharoen, Linda Tan, and Tun Tao Tsai. An algorithm for optimal partitioning of data on an interval. *IEEE Signal Processing Letters*, 12(2):105–108, 2005.

Jeong Min Jeon and Ingrid Van Keilegom. Density estimation for mixed Euclidean and non-Euclidean

data in the presence of measurement error. *Journal of Multivariate Analysis*, 193:105–125, 2023.

Feiyu Jiang, Changbo Zhu, and Xiaofeng Shao. Two-sample and change-point inference for non-euclidean valued time series. *Electronic Journal of Statistics*, 18(1):848–894, 2024.

H. S; Kaplan, O. Salazar Thula, N. Khoss, and M. Zimmer. Nested neuronal dynamics orchestrate a behavioral hierarchy across timescales. *Neuron*, 105: 562–576.e9, 2020. ISSN 0896-6273.

Wilfrid S Kendall. Probability, convexity, and harmonic maps with small image i: uniqueness and fine existence. *Proceedings of the London Mathematical Society*, 3(2):371–406, 1990.

Taha Khan, Lina E Lundgren, Eric Järpe, M Charlotte Olsson, and Pelle Viberg. A Novel Method for Classification of Running Fatigue Using Change-Point Segmentation. *Sensors*, 19(21), 2019.

Rebecca Killick, Paul Fearnhead, and Idris A Eckley. Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, 107(500):1590–1598, 2012.

Solt Kovács, Peter Bühlmann, Housen Li, and Axel Munk. Seeded binary segmentation: a general methodology for fast and optimal changepoint detection. *Biometrika*, 110(1):249–256, 2023.

Chung-Bow Lee. Estimating the number of change points in exponential families distributions. *Scandinavian Journal of Statistics*, 24(2):201–210, 1997.

Lingjun Li and Jun Li. Online change-point detection in high-dimensional covariance structure with application to dynamic networks. *Journal of Machine Learning Research*, 24(51):1–44, 2023.

Z. Lin and H.-G. Müller. Total variation regularized Fréchet regression for metric-space valued data. *The Annals of Statistics*, 49(6):3510–3533, 2021.

Zhenhua Lin, Dehan Kong, and Qiang Sun. Modeling symmetric positive definite matrices with an application to functional brain connectivity. *arXiv preprint arXiv:1907.03385*, 2019.

Haoyang Liu, Chao Gao, and Richard J. Samworth. Minimax rates in sparse, high-dimensional change point detection. *The Annals of Statistics*, 49(2): 1081–1112, 2021.

K. Luxem, P. Mocellin, F. Fuhrmann, J. Kürsch, S. R. Miller, J. J Palop, S. Remy, and P. Bauer. Identifying behavioral structure from deep variational embeddings of animal motion. *Communications Biology*, 5:1267, 2022.

Robert Maidstone, Toby Hocking, Guillem Rigaill, and Paul Fearnhead. On optimal multiple changepoint algorithms for large data. *Statistics and computing*, 27:519–533, 2017.

Gazzoni Marco, Botter Alberto, and Vieira Taian. Surface emg and muscle fatigue: multi-channel approaches to the study of myoelectric manifestations of muscle fatigue. *Physiological measurement*, 38(5): R27, 2017.

A. Olshen, E. Venkatraman, R. Lucito, and M. Wigler. Circular binary segmentation for the analysis of array-based dna copy number data. *Biostatistics*, 5, 2004.

Victor M Panaretos and Yoav Zemel. *An invitation to statistics in Wasserstein space*. Springer Nature, 2020.

Liudmila Pishchagina, Guillem Rigaill, and Vincent Runge. Geometric-based pruning rules for change point detection in multiple independent time series. *arXiv preprint arXiv:2306.09555*, 2023.

Susanna Rampichini, Taian Martins Vieira, Paolo Castiglioni, and Giampiero Merati. Complexity analysis of surface electromyography for assessing the myoelectric manifestation of muscle fatigue: A review. *Entropy*, 22(5):529, 2020.

Stephen Ranshous, Shitian Shen, Danai Koutra, Steve Harenberg, Christos Faloutsos, and Nagiza F Samatova. Anomaly detection in dynamic networks: a survey. *Wiley Interdisciplinary Reviews: Computational Statistics*, 7(3):223–247, 2015.

Gaetano Romano, Guillem Rigaill, Vincent Runge, and Paul Fearnhead. Detecting abrupt changes in the presence of local fluctuations and autocorrelated noise. *Journal of the American Statistical Association*, 117(540):2147–2162, 2022.

Marlene Cristina Neves Rosa, Alda Marques, Sara Demain, Cheryl D Metcalf, and João Rodrigues. Methodologies to assess muscle co-contraction during gait in people with neurological impairment–a systematic literature review. *Journal of Electromyography and Kinesiology*, 24(2):179–191, 2014.

Giulio Rossetti and Rémy Cazabet. Community discovery in dynamic networks: a survey. *ACM computing surveys (CSUR)*, 51(2):1–37, 2018.

Sam Roweis. Constrained Hidden Markov Models. In S Solla, T Leen, and K Müller, editors, *Advances in Neural Information Processing Systems*, volume 12. MIT Press, 1999.

Vincent Runge, Toby Dylan Hocking, Gaetano Romano, Fatemeh Afghah, Paul Fearnhead, and Guillem Rigaill. gfpop: An r package for univariate graph-constrained change-point detection. *Journal of Statistical Software*, 106(6):1–39, 2023.

K. Scherpereel, D. Molinaro, O. Inan, M. Shepherd, and A. Young. A human lower-limb biomechanics and wearable sensors dataset during cyclic and non-cyclic activities. *Scientific Data*, 10(1):1–12, 2023.

Christof Schötz. Nonparametric regression in nonstandard spaces. *Electronic Journal of Statistics*, 16(2): 4679–4741, 2022.

Changhao Shi, Sivan Schwartz, Shahar Levy, Shay Achvat, Maisan Abboud, Amir Ghanayim, Jackie Schiller, and Gal Mishne. Learning disentangled behavior embeddings. In M Ranzato, A Beygelzimer, Y Dauphin, P S Liang, and J Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 22562–22573. Curran Associates, Inc., 2021.

Karl-Theodor Sturm. Probability measures on metric spaces of nonpositive curvature. In *Heat Kernels and Analysis on Manifolds, Graphs, and Metric Spaces*, volume 338, pages 357–390. American Mathematical Society, Paris, France, contempora edition, 2002.

Charles Truong, Laurent Oudre, and Nicolas Vayatis. Selective review of offline change point detection methods. *Signal Processing*, 167:107299, 2020.

Karan Veer and Ravinder Agarwal. Wavelet and short-time fourier transform comparison-based analysis of myoelectric signals. *Journal of Applied Statistics*, 42 (7):1591–1601, 2015.

Nicolas Verzelen, Magalie Fromont, Matthieu Lerasle, and Patricia Reynaud-Bouret. Optimal change-point detection and localization. *The Annals of Statistics*, 51(4):1586–1610, aug 2023.

X. Wang, R. A. Borsoi, and C. Richard. Online Change Point Detection on Riemannian Manifolds With Karcher Mean Estimates. In *Proceedings of the IEEE European Signal Processing (EUSIPCO)*, pages 2033–2037, Helsinki, Finland, 2023a.

X. Wang, R. A. Borsoi, and C. Richard. Nonparametric online change point detection on riemannian manifolds. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 235, pages 50143–50162, 2024.

Xueqin Wang, Jin Zhu, Wenliang Pan, Junhao Zhu, and Heping Zhang. Nonparametric Statistical Inference via Metric Distribution Function in Metric Spaces. *Journal of the American Statistical Association*, pages 1–13, dec 2023b.

C. Weinreb, J. E. Pearl, S. Lin, M. A. M. Osman, L. Zhang, S. Annapragada, E. Conlin, R. Hoffman, S. Makowska, W. F. Gillis, M. Jay, S. Ye, A. Mathis, M. W. Mathis, T. Pereira, S. W. Linderman, and S. R. Datta. Keypoint-moseq: parsing behavior by linking point tracking to pose dynamics, 2 2024a. data from keypoint-moseq.

Caleb Weinreb, Jonah E Pearl, Sherry Lin, Mohammed Abdal Monium Osman, Libby Zhang, Sidharth Annapragada, Eli Conlin, Red Hoffmann, Sofia Makowska, Winthrop F Gillis, et al. Keypoint-moseq: parsing behavior by linking point tracking to pose dynamics. *Nature Methods*, 21(7):1329–1339, 2024b.

Alexander B. Wiltschko, Tatsuya Tsukahara, Ayman Zeine, Rockwell Anyoha, Winthrop F. Gillis, Jeffrey E. Markowitz, Ralph E. Peterson, Jesse Katon, Matthew J. Johnson, and Sandeep Robert Datta. Revealing the structure of pharmacobehavioral space through motion sequencing. *Nature Neuroscience*, 23:1433–1443, 2020.

D. Wu, S. Gundimeda, S. Mou, and C. Quinn. Unsupervised change point detection in multivariate time series. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 238, pages 3844–3852, 10 2024.

Chao Zhang, Piotr Kokoszka, and Alexander Petersen. Wasserstein autoregressive models for density time series. *Journal of Time Series Analysis*, 43(1):30–52, 2022.

Nancy R. Zhang and David O. Siegmund. A modified bayes information criterion with applications to the analysis of comparative genomic hybridization data. *Biometrics*, 63(1):22–32, 2007.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]. Sections 2.2 and 3.2
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]. Section 3.1
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]. Section 4.2
   (b) Complete proofs of all theoretical results. [Yes]. See Appendix B
   (c) Clear explanations of any assumptions. [Yes]. Section 4.2

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Yes]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes]
   (d) Information about consent from data providers/curators. [Yes]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A Proof of Proposition 1

The dynamic programming update rule for the time series $y_1, \dots, y_T$, at time $t+1$, given the last state value c_k, is given for $k \in \{1, \dots, V\}$ by:

$$Q_k(t+1) = \min_{i=1,\dots,V} \left\{ Q_i(t) + \beta \mathbf{1}_{\{i \neq k\}} \right\} + d^2(y_{t+1}, c_k)\,, \tag{17}$$

with the initial condition $Q_k(0) = 0$. This update rule returns the minimum cost $Q(T) = \min_k Q_k(T)$. To state this result, we come back to the definition of the optimal cost defined in Equation 7 of the main document, isolating the last point y_{t+1} forced to be associated to state value c_k:

$$Q_k(t+1) = \min_{\boldsymbol{p} \in \{1,\dots,V\}^t} \left[\sum_{u=1}^{t} d^2(y_u, c_{p(u)}) + d^2(y_{t+1}, c_k) + \beta \sum_{u=2}^{t} \mathbf{1}_{\{p(u-1) \neq p(u)\}} + \beta \mathbf{1}_{\{p(t) \neq k\}} \right].$$

Two cases can be distinguished, depending on the state value at time step t:

$$Q_k(t+1) = \begin{cases} Q_k(t) + d^2(y_{t+1}, c_k) & \text{if } p(t) = k \\ \min_{\boldsymbol{p} \in \{1,\dots,V\}^t} \left[\sum_{u=1}^{t} d^2(y_u, c_{p(u)}) + \beta \sum_{u=2}^{t} \mathbf{1}_{\{p(u-1) \neq p(u)\}} \right] + d^2(y_{t+1}, c_k) + \beta & \text{if } p(t) \neq k\,. \end{cases}$$

In the second case, we are back to the initial problem. The minimization can be expressed as $\min_{k \in 1,\dots V} Q_k(t)$: the minimal value is obtained at one of the V available state values, which proves the recursion.
The path $\hat{\boldsymbol{p}}$ that solves the penalized optimization problem in Equation 7 is obtained through the backtracking procedure, which is a fundamental step in dynamic programming. At each time step t and for each possible state index i, we store the optimal "parent" state used in the minimization in (17). By identifying the argmin in k for $\{Q_k(t+1), k = 1, \dots, V\}$, we determine the final state $\hat{p}(T)$. Since the best index is saved during the update step, we can iteratively trace back to find $\hat{p}(T-1)$, and so forth, ultimately recovering the initial state $\hat{p}(1)$.

B Proof of Section 4

Proofs in Section 4 make use of the properties of probability measures defined on Hadamard spaces which we now define. Let $(\mathcal{H}, d)$ be a Hadamard space, and denote by $\mathcal{P}^2(\mathcal{H})$ the set of probability measures p on $(\mathcal{H}, \mathcal{B}(\mathcal{H}))$ with separable support, where $\mathcal{B}(\mathcal{H})$ is the Borel σ-algebra defined on $\mathcal{H}$, such that $\int d^2(x,y)p(\mathrm{d}y) < \infty$ for all $x \in \mathcal{H}$.

Definition 3. *Fréchet mean and variance of a random variable]. Let Y be a $\mathcal{H}$-valued random variable with distribution function $p_Y \in \mathcal{P}^2(\mathcal{H})$.*

- *The expected value of Y, denoted $\mathbb{E}Y \in \mathcal{H}$, is defined as its Fréchet mean: $\mathbb{E}Y := \mu(p_Y) = \arg\min_\mu \int d^2(Y, \mu) p_Y(\mathrm{d}y)$;*
- *The variance of Y, denoted $\mathbb{V}Y$, is defined by $\mathbb{V}Y := \int d^2(y, \mathbb{E}Y) p_Y(\mathrm{d}y)$. Note that $\mathbb{V}Y < +\infty$ since $p_Y \in \mathcal{P}^2(\mathcal{H})$.*

Proposition 2 (Variance and Reverse Variance Inequalities, (Sturm, 2002, Proposition 4.4))**.** *Let $p \in \mathcal{P}^2(\mathcal{H})$ be a probability measure with Fréchet mean $\mu(p)$, and let Y be an $\mathcal{H}$-valued random variable. For all $z \in \mathcal{H}$, the following hold:*
1. Variance Inequality:

$$\int d^2(y,z)p(\mathrm{d}y) - d^2(z, \mu(p)) \geq \int d^2(y, \mu(p))p(\mathrm{d}y) \geq 0,$$

Similarly,

$$\mathbb{E}d^2(Y,z) - d^2(\mathbb{E}Y, z) \geq \mathbb{E}d^2(Y, \mathbb{E}Y) = \mathbb{V}Y. \tag{18}$$

2. Reverse Variance Inequality: For any $q \in \mathcal{P}^2(\mathcal{H})$,

$$\int \left[d^2(z,x) - d^2(z, \mu(p)) - d^2(\mu(p), x) \right] \mu(\mathrm{d}x) \leq \frac{2\kappa^2}{3} \int \left[d^4(z, \mu(p)) + d^4(\mu(p), x) \right] p(\mathrm{d}x), \tag{19}$$

where $-\kappa^2 \leq curv(\mathcal{H}, d)$ is the global curvature bound of the Hadamard space. In terms of random variables, we also write

$$\mathbb{E}d^2(Y, z) - \mathbb{E}d^2(Y, \mathbb{E}Y) \leq \frac{2\kappa^2}{3}\left[d^4(z, \mathbb{E}Y) + \mathbb{E}d^4(Y, \mathbb{E}Y)\right] + d^2(\mathbb{E}Y, z). \tag{20}$$

In particular, we see that Equation (12) in Assumption 3 of the main document holds when, for $z = c_k$:

$$\frac{2\kappa^2}{3}\left[d^4(c_k, c_k^*) + \mathbb{E}d^4(Y_t, c_k^*)\right] \leq \gamma\,.$$

B.1 Proof of Theorem 1

Lemma 1. *For the update of the alternating algorithm 1, the size of the decrease in the objective function is large or equal to the square of the distance between current value $\boldsymbol{c}^{(i)}$ and new value $\boldsymbol{c}^{(i+1)}$:*

$$\Phi^{(i)} - \Phi^{(i+1)} \geq d^2(\boldsymbol{c}^{(i+1)}, \boldsymbol{c}^{(i)})\,. \tag{21}$$

Proof.

$$\begin{aligned}
\Phi^{(i)} - \Phi^{(i+1)} &= \Phi(\boldsymbol{c}^{(i)}, \boldsymbol{p}^{(i)}) - \Phi(\boldsymbol{c}^{(i+1)}, \boldsymbol{p}^{(i+1)}) \\
&\geq \Phi(\boldsymbol{c}^{(i)}, \boldsymbol{p}^{(i+1)}) - \Phi(\boldsymbol{c}^{(i+1)}, \boldsymbol{p}^{(i+1)}) \\
&\geq d^2(\boldsymbol{c}^{(i+1)}, \boldsymbol{c}^{(i)})\,,
\end{aligned}$$

where the first inequality is by definition of $\boldsymbol{p}^{(i+1)}$ (Equation 10) and the second one is from the variance inequality (Proposition 2). Indeed:

$$\begin{aligned}
\Phi(\boldsymbol{c}^{(i)}, \boldsymbol{p}^{(i+1)}) - \Phi(\boldsymbol{c}^{(i+1)}, \boldsymbol{p}^{(i+1)}) &= \sum_{t=1}^{T} d^2(y_t, c^{(i)}_{p(t)^{(i+1)}}) - \sum_{t=1}^{T} d^2(y_t, c^{(i+1)}_{p(t)^{(i+1)}}) \\
&= \sum_{j=1}^{V}\sum_{t\in S_j} d^2(y_t, c_j^{(i)}) - \sum_{j=1}^{V}\sum_{t\in S_j} d^2(y_t, c_j^{(i+1)}) \\
&= \sum_{j=1}^{V} |S_j| \left[\int d^2(y, c_j^{(i)}) p_j(dy) - \int d^2(y, c_j^{(i+1)}) p_j(dy)\right] \\
&\geq \sum_{j=1}^{V} |S_j| d^2(c_j^{(i+1)}, c_j^{(i)}) \geq \sum_{j=1}^{V} d^2(c_j^{(i+1)}, c_j^{(i)}) = d^2(\boldsymbol{c}^{(i+1)}, \boldsymbol{c}^{(i)})
\end{aligned}$$

where $p_j(dy) = 1/|S_j| \sum_{t\in S_j} \delta_{y_t}$ is a probability measure that is an empirical distribution of y_t's for $t \in S_j$ where $S_j = \{t : s_t^{(i+1)} = j\}$, for $j \in \{1, \ldots, V\}$. We suppress the dependence on $i+1$ in S_j and p_j for ease of notation. The inequality follows from Proposition 2 and the fact that $c_j^{(i+1)}$ is the Fréchet mean for $p_j(dy)$. If $|S_j|$ vanishes, then $d^2(c_j^{(i)}, c_j^{(i+1)}) = 0$, there is no state update. This makes the last inequality true. □

Proof of Theorem 1

Proof. The limit value $\Phi^{(\infty)}$ of the sequence $(\Phi^{(i)})_{i\geq 1}$ is well-defined since it is a decreasing sequence (Lemma 1 in Appendix) bounded from below as $\inf \Phi > -\infty$. Suppose that Algorithm 1 executes κ iterations without terminating, then $d^2(\boldsymbol{c}^{(i+1)}, \boldsymbol{c}^{(i)}) > \epsilon$ for all $i = 1, \ldots, \kappa$. Then using Lemma 1 (derived from the variance inequality in Hadamard spaces) and a telescoping sum argument yields:

$$\begin{aligned}
\Phi^{(1)} - \Phi^{(\infty)} \geq \Phi^{(1)} - \Phi^{(\kappa+1)} &= \sum_{i=1}^{\kappa} \left(\Phi^{(i)} - \Phi^{(i+1)}\right) \\
&\geq \sum_{i=1}^{\kappa} d^2(\boldsymbol{c}^{(i+1)}, \boldsymbol{c}^{(i)}) > \kappa\epsilon\,,
\end{aligned}$$

which contradicts the definition of κ. □

B.2 Proof of Theorem 2

Define for any $\theta \in \mathcal{H}$, the true successive levels $\boldsymbol{\theta}^\star = (\theta_t^\star)_{t=1}^T$ and their closest projections onto $\mathcal{C}$, $P(\boldsymbol{\theta}^\star) = (P(\theta_t^\star))_{t=1}^T$, the following quantities for all $t \in \{1, \dots, T\}$:

$$\begin{cases} \Delta\mathbf{D}_t^2(\theta) & := d^2(Y_t, \theta) - d^2(Y_t, P(\theta_t^\star))\,, \\ \Delta\mathbf{d}_t^2(\theta) & := \mathbb{E}\left[\Delta\mathbf{D}_t^2(\theta)\right]\,, \\ \Delta\varepsilon_t(\theta) & := \Delta\mathbf{D}_t^2(\theta) - \Delta\mathbf{d}_t^2(\theta)\,. \end{cases}$$

We also have (as in Assumption 2 we defined $\varepsilon_t(\theta) := d^2(Y_t, \theta) - \mathbb{E}\left[d^2(Y_t, \theta)\right]$):

$$\begin{aligned} \Delta\varepsilon_t(\theta) &= d^2(Y_t, \theta) - \mathbb{E}\left[d^2(Y_t, \theta)\right] - \left(d^2(Y_t, P(\theta_t^\star)) - \mathbb{E}\left[d^2(Y_t, P(\theta_t^\star))\right]\right)\,, \\ &= \varepsilon_t(\theta) - \varepsilon_t(P(\theta_t^\star))\,. \end{aligned} \tag{22}$$

The sequence $\hat{\boldsymbol{\theta}} = (c_{\hat{p}(t)})_{t=1}^T$ is an Hadamard estimator (see Equation 7 in the main document) where $\hat{\boldsymbol{p}}$ is a minimizing path for ϕ, and $P(\boldsymbol{\theta}^\star) = (c_{p^*(t)})_{t=1}^T$ its closest projection on accessible states in $\mathcal{C}$. We then have:

$$\phi(\hat{\boldsymbol{p}}) - \phi(\boldsymbol{p}^\star) = \sum_{t=1}^T \Delta\mathbf{d}_t^2(c_{p(t)}) + \sum_{t=1}^T \Delta\varepsilon_t(c_{p(t)}) + \beta\hat{K} - \beta K^\star \leq 0\,. \tag{23}$$

Notice that the number of changes in $\boldsymbol{p}^\star$ is also $K^\star$ due to Assumption 3. The projected path would have a smaller number of changes only if some of the consecutive states are projected to the same available states $P(c_k^\star) = P(c_{k+1}^\star) = c_k$. However in this case Assumption 3 is in contradiction with the property of the metric (2) which requires $d^2(c_k, c_k^\star) + d^2(c_k, c_{k+1}^\star) \geq \frac{1}{2}d^2(c_k^\star, c_{k+1}^\star)$.

We will use this inequality (23) repeatedly in the proof.

Lemma 2. *Define the partial sums $S_{s..t}(\theta) := \sum_{i=s}^t \Delta\varepsilon_i(\theta)$. Then, under Assumption 2, the following holds true, for any $\delta > 0$:*

$$\mathbb{P}\left(\max_{\theta\in\mathcal{C}} \max_{1\leq s<t\leq T} |S_{s..t}(\theta)| \geq \delta\right) \leq 4\sigma^2 VT/\delta^2. \tag{24}$$

Proof. First notice that $\Delta\varepsilon_t(\theta) = \varepsilon_t(\theta) - \varepsilon_t(P(\theta_t^\star))$. Therefore, under Assumption 2, for any (t, t') with $t \neq t'$, and any (θ, θ'), the random variables $\Delta\varepsilon_t(\theta)$ and $\Delta\varepsilon_{t'}(\theta')$ are independent. By Cauchy-Schwarz, for $\theta \in \mathcal{C}$, $\mathbb{E}\Delta\varepsilon_t(\theta)^2 = \mathbb{E}\varepsilon_t(\theta)^2 + \mathbb{E}\varepsilon_t(P(\theta_t^\star))^2 + 2\mathbb{E}\left[-\varepsilon_t(\theta)\varepsilon_t(P(\theta_t^\star))\right] \leq 4\max_{\theta\in\mathcal{C}} \mathbb{E}\varepsilon_t(\theta)^2$, so the variances for variables $\Delta\varepsilon_t(\theta)$ are finite and bounded above uniformly over $t \in \{1, \dots, T\}$ and $\theta \in \mathcal{C}$, i.e.

$$\max_{t=1,\dots,T} \max_{\theta\in\mathcal{C}} \mathbb{E}\Delta\varepsilon_t(\theta)^2 < \sigma^2 < \infty. \tag{25}$$

By Kolmogorov's inequality,

$$\mathbb{P}\left(\max_{1\leq t\leq T} |S_{1..t}(\theta)| \geq \delta\right) \leq \sigma^2 T/\delta^2. \tag{26}$$

Applying the union bound yields:

$$\mathbb{P}\left(\max_{\theta\in\mathcal{C}} \max_{1\leq t\leq T} |S_{1..t}(\theta)| \geq \delta\right) \leq V\sigma^2 T/\delta^2\,. \tag{27}$$

Now, notice that for all s, t such that $2 \leq s < t \leq T$:

$$|S_{s..t}(\theta)| = |S_{1..t}(\theta) - S_{1..(s-1)}(\theta)| \leq 2 \max_{1\leq t'\leq T} |S_{1..t'}(\theta)|. \tag{28}$$

As a result:

$$\begin{aligned} \mathbb{P}\left(\max_{\theta\in\mathcal{C}} \max_{1\leq s<t\leq T} |S_{s..t}(\theta)| \geq \delta\right) &\leq \mathbb{P}\left(\max_{\theta\in\mathcal{C}} \max_{1\leq t\leq T} |S_{1..t}(\theta)| \geq \delta/2\right) \\ &\leq 4\sigma^2 VT/\delta^2. \end{aligned} \tag{29}$$

□

We can now prove Theorem 2. Define the event $\mathcal{E} := \{H(\{t_k^\star\}, \{\hat{t}_k\})/T \geq \delta\}$, as well as the mutually exclusive events $\Omega_= := \{\hat{K} = K^\star\}$, $\Omega_< := \{\hat{K} < K^\star\}$ and $\Omega_> := \{\hat{K} > K^\star\}$. To prove Equation 15, we will bound the probabilities of $\Omega_<$ and $\Omega_>$ and show that they converge to zero. To prove Equation 16, we will bound $\mathbb{P}(\mathcal{E} \cap \Omega_=)$.

Case $\hat{K} < K^\star$ We consider that $\hat{K} < K^\star$ and will derive from it useful implications in order to upper bound the probability of event $\Omega_<$. Let $L := \lceil T\delta_{\min}/3 \rceil \geq 1$ (T is large). Since $\hat{K} < K^\star$, there exists a $t^\star \in \{t_k^\star\}_{k=1}^{K^\star}$ such that $\mathbf{I} = (t^\star - L)..(t^\star + L - 1)$ does not contain any element of $\{\hat{t}_k\}_{k=1}^{\hat{K}}$. (Otherwise, since there are $K^\star$ such intervals and they are disjoint, this would imply that there are at least $K^\star$ elements in $\{\hat{t}_k\}_{k=1}^{\hat{K}}$, which contradicts the assumption that $\hat{K} < K^\star$.) As a result, on this interval $(t^\star - L)..(t^\star + L - 1)$, $\hat{\boldsymbol{\theta}}$ is constant, equal to, say, $\hat{c}$. On $(t^\star - L)..(t^\star - 1)$, $\boldsymbol{\theta}^\star$ is equal to $c_k^\star$ for a certain k, and $c_{k+1}^\star$ on $t^\star..(t^\star + L - 1)$. We get the following sequence of lower bounds:

$$
\begin{aligned}
\sum_{t=1}^{T} \Delta \mathbf{d}_t^2(\hat{\theta}_t) &\geq \sum_{t=1}^{T} d^2(\hat{\theta}_t, \theta_t^\star) + \sum_{t=1}^{T} \mathbb{E}\left[d^2(Y_t, \theta_t^\star) - d^2(Y_t, P(\theta_t^\star))\right] \quad \text{(using variance inequality (18))} \\
&\geq -\gamma T + \sum_{t \in \mathbf{I}} \Big(d^2(\hat{\theta}_t, \theta_t^\star) - d^2(\theta_t^\star, P(\theta_t^\star)) \Big) + \sum_{t \notin \mathbf{I}} \Big(d^2(\hat{\theta}_t, \theta_t^\star) - d^2(\theta_t^\star, P(\theta_t^\star)) \Big) \quad \text{(using } \textit{Assumption } 3) \\
&\geq -\gamma T + \sum_{t \in \mathbf{I}} \Big(d^2(\hat{\theta}_t, \theta_t^\star) - d^2(\theta_t^\star, P(\theta_t^\star)) \Big) \quad \text{(as } d^2(\theta, \theta_t^\star) - d^2(P(\theta_t^\star), \theta_t^\star) \geq 0 \,, \forall \theta \in \mathcal{C} \,, \, \forall t) \\
&= -\gamma T + L d^2(\hat{c}, c_k^\star) + L d^2(\hat{c}, c_{k+1}^\star) - \Big(L d^2(P(c_k^\star), c_k^\star) + L d^2(P(c_{k+1}^\star), c_{k+1}^\star) \Big) \\
&\geq -\gamma T + \frac{L}{2} \Big(d^2(c_k^\star, c_{k+1}^\star) - 2(d^2(P(c_k^\star), c_k^\star) + d^2(P(c_{k+1}^\star), c_{k+1}^\star)) \Big) \quad \text{(property of the metric (1))} \\
&\geq -\gamma T + \frac{L}{4} d^2(c_k^\star, c_{k+1}^\star) \quad \text{(using } \textit{Assumption } 3) \\
&\geq T\Big(\frac{\Delta_{\min}\delta_{\min}}{12} - \gamma\Big) > 0 \quad \text{(using } \textit{Assumption } 1) \,.
\end{aligned}
\tag{30}
$$

Now, recall that $\hat{\boldsymbol{\theta}}$ is constant on each interval $(\hat{t}_k + 1)..\hat{t}_{k+1}$. As a result,

$$
\left| \sum_{t=1}^{T} \Delta \varepsilon_t(\hat{\theta}_t) \right| = \left| \sum_{k=0}^{\hat{K}} \sum_{t=\hat{t}_k+1}^{\hat{t}_{k+1}} \Delta \varepsilon_t(\hat{\theta}_t) \right| \leq (\hat{K} + 1) \max_{\theta \in \mathcal{C}} \max_{s,t} |S_{s..t}(\theta)| \leq (K^\star + 1) \max_{\theta \in \mathcal{C}} \max_{s,t} |S_{s..t}(\theta)| \,. \tag{31}
$$

From (23) we get:

$$
0 \geq \phi(\hat{\boldsymbol{p}}) - \phi(\boldsymbol{p}^\star) \geq \sum_{t=1}^{T} \Delta \mathbf{d}_t^2(\hat{\theta}_t) - \left| \sum_{t=1}^{T} \Delta \varepsilon_t(\hat{\theta}_t) \right| - \beta K^\star \,,
$$

and thus using (30) and (31) yields:

$$
-(K^\star + 1) \max_{\theta \in \mathcal{C}} \max_{s,t} |S_{s..t}(\theta)| + T\Big(\frac{\Delta_{\min}\delta_{\min}}{12} - \gamma\Big) - \beta K^\star \leq \phi(\hat{\boldsymbol{p}}) - \phi(\boldsymbol{p}^\star) \leq 0 \,,
$$

or in term of probability of the event $\Omega_<$ (writing $\beta = \lambda T$):

$$
\begin{aligned}
\mathbb{P}\left(\hat{K} < K^\star\right) &\leq \mathbb{P}\left(T\left[\frac{\Delta_{\min}\delta_{\min}}{12} - \gamma - \lambda K^\star \right] \leq (K^\star + 1) \max_{\theta \in \mathcal{C}} \max_{s,t} |S_{s..t}(\theta)| \right) \\
&\leq \frac{4\sigma^2 V T (K^\star + 1)^2}{T^2 \left[\frac{\Delta_{\min}\delta_{\min}}{12} - \gamma - \lambda K^\star \right]^2} \quad \text{(by Lemma 2)} \\
&\leq \frac{C_<}{T} \quad \text{where } C_< := \frac{4\sigma^2 V (K^\star + 1)^2}{\left[\frac{\Delta_{\min}\delta_{\min}}{12} - \gamma - \lambda K^\star \right]^2} .
\end{aligned}
\tag{32}
$$

Note that the quantity $\Delta_{\min}\delta_{\min}/12 - \lambda K^\star$ is positive thanks to the assumption on λ. As a result, $\mathbb{P}\left(\hat{K} < K^\star\right)$ converges to 0 as $T \to \infty$.

Case $\hat{K} > K^\star$ We consider that $\hat{K} > K^\star$ and will derive from it useful implications to upper bound the probability of the event $\Omega_>$. From (23), we derive:

$$\beta(\hat{K} - K^\star) \leq (\hat{K}+1) \max_{\theta\in\mathcal{C}} \max_{s,t} |S_{s..t}(\theta)| \,,$$

and using relations:

$$\frac{\hat{K}+1}{K^\star+2} = (\hat{K}+1)\Big(1 - \frac{K^\star+1}{K^\star+2}\Big) \leq (\hat{K}+1)\Big(1 - \frac{K^\star+1}{\hat{K}+1}\Big) = \hat{K} - K^\star \,,$$

we obtain

$$\lambda T/(K^\star+2) \leq \max_{\theta\in\mathcal{C}} \max_{s,t} |S_{s..t}(\theta)| \,. \tag{33}$$

Thus:

$$\begin{aligned}
\mathbb{P}\left(\hat{K} > K^\star\right) &\leq \mathbb{P}\left(\lambda T/(K^\star+2) \leq \max_{\theta\in\mathcal{C}} \max_{s,t} |S_{s..t}(\theta)|\right) \\
&\leq \frac{4\sigma^2 V T}{(\lambda T/(K^\star+2))^2} \\
&\leq \frac{C_>}{T} \quad \text{where } C_> := \frac{4\sigma^2 V (K^\star+2)^2}{\lambda^2}.
\end{aligned} \tag{34}$$

We proved that $\mathbb{P}(\hat{K} \neq K^\star) = \mathbb{P}(\hat{K} < K^\star) + \mathbb{P}(\hat{K} > K^\star) \leq (C_< + C_>)/T$ and therefore $\lim_{T\to\infty} \mathbb{P}(\hat{K} \neq K^\star) = 0$, hence $\lim_{T\to\infty} \mathbb{P}(\hat{K} = K^\star) = 1$, which is the result to prove Equation 15.

Case $\hat{K} = K^\star$ We consider that $\hat{K} = K^\star$ and that the event $\mathcal{E}$ is true. Since, for any $\delta < \delta'$,

$$\mathbb{P}\left(H(\{t_k^\star\}, \{\hat{t}_k\})/T \geq \delta'\right) \leq \mathbb{P}\left(H(\{t_k^\star\}, \{\hat{t}_k\})/T \geq \delta\right), \tag{35}$$

we can restrict ourselves to $\delta < \delta_{\min}/3$, without loss of generality (as we want to upper bound this probability). Define $L := \lceil T\delta_{\min}/3 \rceil$. Necessarily there exists $t^\star \in \{t_k^\star\}$ such that the interval $(t^\star - L)..(t^\star + L - 1)$ does not contain any estimated change point (by definition of the Hausdorff distance), i.e. $(t^\star - L)..(t^\star + L - 1) \cap \{\hat{t}_k\}_k = \emptyset$. Otherwise, this would contradict the assumption that we are on $\mathcal{E} \cap \Omega_=$ and that $\delta < \delta_{\min}/3$. On the interval $(t^\star - L)..(t^\star + L - 1)$, the estimated signal $\hat{\boldsymbol{\theta}}$ is constant equal to, says, $\hat{c}$. On $(t^\star - L)..t^\star$, $\boldsymbol{\theta}^\star$ is equal to $c_k^\star$ for a certain k, and $c_{k+1}^\star$ on $t^\star..(t^\star + L - 1)$. Similarly to the case $\hat{K} < K^\star$ (see the sequence of inequalities in (30)), this yields:

$$\sum_{t=1}^{T} \Delta \mathbf{d}_t^2(\hat{\theta}_t) \geq T\left(\frac{\Delta_{\min}\delta_{\min}}{12} - \gamma\right). \tag{36}$$

Similarly to (31), we also have:

$$\left|\sum_t \Delta\varepsilon_t(\hat{\theta}_t)\right| \leq (K^\star+1) \max_{\theta\in\mathcal{C}} \max_{s,t} |S_{s..t}(\theta)| \,. \tag{37}$$

Combining (23), (36) and (37) yields:

$$\mathbb{P}\left(\mathcal{E} \cap \Omega_=\right) \leq \frac{C_=}{T\delta^2} \quad \text{where } C_= := \frac{4\sigma^2 V (K^\star+1)^2}{\left[\frac{\Delta_{\min}\delta_{\min}}{12} - \gamma\right]^2}. \tag{38}$$

Finally,

$$\begin{aligned}
\mathbb{P}\left(H(\{t_k^\star\}, \{\hat{t}_k\})/T \geq \delta\right) &\leq \mathbb{P}\left(\mathcal{E} \cap \Omega_=\right) + \mathbb{P}\left(\Omega_<\right) + \mathbb{P}\left(\Omega_>\right) \\
&\leq \frac{C_=}{\delta^2 T} + \frac{C_<}{T} + \frac{C_>}{T} \,,
\end{aligned} \tag{39}$$

which is equivalent to Equation 16 for appropriate constants C_H and C'_H.

Remark 1. *Note that all parameters from the assumptions of Section 4 ($\sigma, \Delta_{\min}, \delta_{\min}$), as well as $K^\star$ are allowed to grow with T, but not too fast, to ensure that the relevant probabilities vanish. For example, if only σ^2 grows with T we could require $\sigma^2(T)/T \to 0$ to have $\mathbb{P}(\hat{K} < K^\star) = \mathbb{P}(\hat{K} > K^\star) \to 0$.*

C Connection to HMM

The optimization problem defined in (5) has a statistical interpretation. Namely, $\hat{\boldsymbol{\theta}}$ can be interpreted as a maximum likelihood estimator for a suitably chosen probabilistic structure of an HMM.

A HMM models a sequence of observations $\boldsymbol{y} = (y_1, \dots, y_T) \subset \mathcal{H}$ with an unobserved ("hidden") sequence of states $\boldsymbol{p} = (p_1, \dots, p_T) \subset [\![1, V]\!]$ and an initial state probability. Formally,

(i) $\forall (t, i) \in [\![1, T]\!] \times [\![1, V]\!]$, the observations $\boldsymbol{y}$ are conditionally independent, i.e. $\mathbb{P}(y_t = y | p_t, p_{t-1}, \dots, p_1) = \mathbb{P}(y_t = y | p_t)$;

(ii) the state sequence is a Markov chain, i.e. $\mathbb{P}(p_t = j | p_{t-1} = i_{t-1}, p_{t-2} = i_{t-2}, \dots, p_1 = i_1) = \mathbb{P}(p_t = j | p_{t-1} = i_{t-1})$ for any $t \in [\![1, T]\!]$ and sequence $(i_1, \dots, i_{t-1}) \subset [\![1, V]\!]$;

(iii) the initial distribution of the state p_1 is $\mathbb{P}(p_1 = i) = \pi(i)$ for a given function $\pi : [\![1, V]\!] \to [0, 1]$.

The function $f(\cdot|i) := \mathbb{P}(\cdot | p_t = i)$ is called the emission probability at state i; the coefficients $a_{ij} := \mathbb{P}(p_t = j | p_{t-1} = i)$ are the transition probabilities; the coefficients $\pi(i)$ are the initial state probabilities. A HMM is completely specified by the $f(\cdot|i)$, a_{ij} and $\pi(i)$, for all $(i, j) \subset [\![1, V]\!]$.

Lemma 3. *Assume the following HMM:*

- *The emission probability at state i is such that its density is $f(y|i) \propto e^{-d^2(y, c_i)}$ for a given center $c_i \in \mathcal{H}$.*
- *The transition probabilities are given by*

$$a_{ij} = \begin{cases} \frac{e^{-\beta}}{1+(V-1)e^{-\beta}} & \text{if } j \neq i \\ \frac{1}{1+(V-1)e^{-\beta}} & \text{if } j = i \end{cases} \tag{40}$$

  for a certain parameter $\beta > 0$.

- *The initial state probabilities are uniform, i.e. $\pi(i) = 1/V$ for all i.*

Then our method `HOP` *(namely the segmentation step only), with centers $\mathcal{C} = \{c_1, \dots, c_V\} \subset \mathcal{H}$ and penalty β, computes the most probable sequence of states of this HMM. This sequence is returned by the so-called Viterbi algorithm in the HMM literature.*

Proof. By definition, the most probable sequence of states, $\hat{\boldsymbol{p}}$ is given by $\arg\max_{\boldsymbol{p} \in [\![1,V]\!]^T} \mathbb{P}(\boldsymbol{p}|\boldsymbol{y})$. We have the following equality:

$$\begin{aligned}
\arg\min_{\boldsymbol{p}} \phi(\boldsymbol{p}) &= \arg\max_{\boldsymbol{p} \in [\![1,V]\!]^T} e^{-\phi(\boldsymbol{p})} \\
&= \arg\max_{\boldsymbol{p} \in [\![1,V]\!]^T} e^{-\sum_{t=1}^T d^2(y_t, c_{p_t}) - \beta \sum_{t=2}^T \mathbf{1}_{\{p_{t-1} \neq p_t\}}} \\
&= \arg\max_{\boldsymbol{p} \in [\![1,V]\!]^T} \prod_{t=1}^T f(y_t|p_t) \prod_{t=2}^T a_{p_{t-1} p_t} \quad \text{using formulae in Lemma 3} \\
&= \arg\max_{\boldsymbol{p} \in [\![1,V]\!]^T} f(\boldsymbol{y}|\boldsymbol{p}) \mathbb{P}(\boldsymbol{p}) \quad \text{using conditions (i) and (ii)} \\
&= \arg\max_{\boldsymbol{p} \in [\![1,V]\!]^T} \mathbb{P}(\boldsymbol{p}|\boldsymbol{y}) f(\boldsymbol{y}) \quad \text{using Bayes' theorem} \\
&= \arg\max_{\boldsymbol{p} \in [\![1,V]\!]^T} \mathbb{P}(\boldsymbol{p}|\boldsymbol{y}) \,.
\end{aligned} \tag{41}$$

Notice that for the transition probabilities in (41), we use this proportionality:

$$\mathbb{P}(\boldsymbol{p}) = \pi(p_1) \prod_{t=2}^T a_{p_{t-1} p_t} \propto e^{-\beta \sum_{t=2}^T \mathbf{1}_{\{p_{t-1} \neq p_t\}}} \,,$$

as the product of the denominators in (40) is independent of the chosen path $\boldsymbol{p}$, and the initial probability is uniformly distributed (see condition (iii)). □

D Constrained Inference

In this section, we describe how to enforce constraints on the transitions between states with `HOP`. These constraints are enforced by defining a binary transition matrix $B \in \{0,1\}^{V \times V}$ where V is the number of states. The entry $B(i,j) = 1$ indicates that a transition from state i to state j is allowed. As a result, the optimal path will belong to the following set, a subset of $\{1,\dots,V\}^T$:

$$\left\{ \boldsymbol{p} \in \{1,\dots,V\}^T : B(p_{t-1}, p_t) = 1\,,\, t = 2,\dots,T \right\}.$$

The transition constraints can be incorporated via B in the update rule Equation 8 and applied in Algorithm 1 of the main paper. We now define the constrained update rule, which can be used in Algorithm 1 for any given binary transition matrix.

$$Q_k(t+1) = \min_{i:B(k,i)=1} \left\{ Q_i(t) + \beta \mathbf{1}_{\{i \neq k\}} \right\} + d^2(y_{t+1}, c_k)\,,$$

Two common structural assumptions of interest are cyclic and epidemic patterns of change. In the cyclic setting, transitions follow a specific order and repeat in a loop. The transition matrix for the cyclic setting is given by $B(i,i) = 1$, $B(i,i+1) = 1$ for $i = 1,\dots V-1$, $B(i,1) = 1$

$$B_{\text{cyclic}} = \begin{pmatrix} 1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 1 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \cdots & 1 \end{pmatrix}.$$

We use this constraint to model time series in the application of Section 6.1.

The epidemic setting assumes that the state sequence alternates between a baseline state ("normal state") and other states. Assuming that State 1 is the baseline state, the transition matrix is given by $B(i,i) = 1$, for $i = 1,\dots,V$, $B(i,1) = 1$ and $B(1,i) = 1$ for $i = 1,\dots,V$.

$$B_{\text{epidemic}} = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 0 & \cdots & 0 \\ 1 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \cdots & 1 \end{pmatrix}.$$

E Experiments

E.1 Supplementary simulation: Influence of the number of states V

The number of states V is a user-defined parameter of our method `HOP`. In our experiments, we choose V to be significantly larger than the maximum number of change points. We then split the signal into segments of the same length and compute the Fréchet mean on each segment to form the set of states $\mathcal{C}$. As a result, there is a good chance that on each true segment, there will be one state that approximates the signal well. In this experiment we show that if V is below a certain value, `HOP` is less accurate.

Data We restrict our study to generated signals with 1000 observations and 8 change points.

Results Detection accuracy is worse when the number of states V is lower than the number of segments. This is because Assumption 3 is no longer verified: a single state approximates two consecutive segments of the true signal. The F-score is optimal once we have a good approximation of each segment mean (when V reaches 9). Note that the plateau starts when V equals the number of segments. This is undoubtedly an artifact of our simulation procedure, where, on average, segments have equal length. In signals with segments of segments of sensibly different sizes, we recommend increasing V to 5 or 10 times the maximum number of changes.

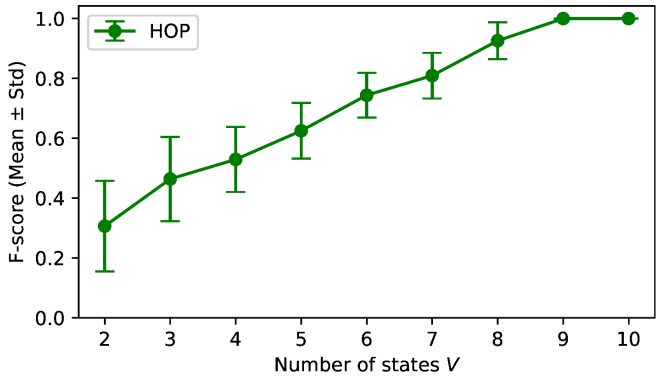
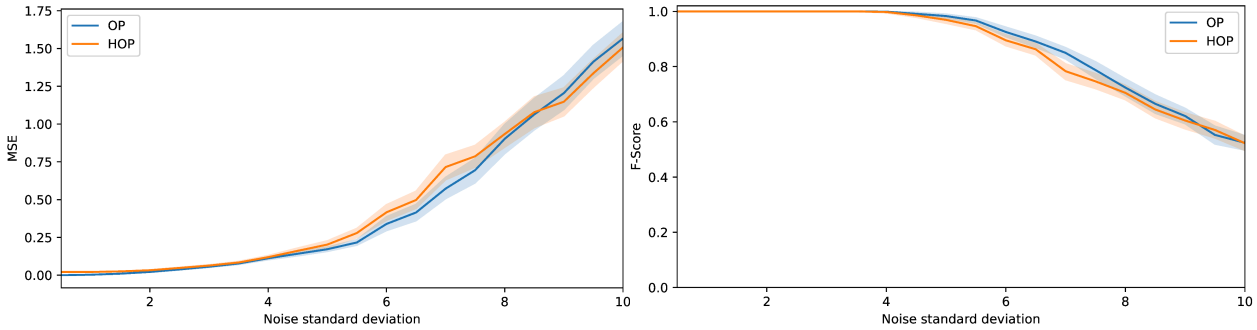


Figure 5: Results on the synthetic data set: F-score with respect to the number of states V.



(a) MSE (b) F-score

Figure 6: Performance comparison between HOP and OP, using the MSE and F-Score (Section E.2).

E.2 Supplementary simulation: how close to Optimal Partitioning is HOP?

This experiment shows that, on $\mathcal{H} = \mathbb{R}^D$, our method HOP finds a solution that is close to the exact Optimal Partitioning (OP) algorithm, with respect to the MSE and F-score.

Data. We generate 100 signals as follows. They all have length $T = 5000$ and dimension $D = 10$. For a signal $(u_t)_{t=1}^{5000}$, a random number K of change point is uniformly chosen between 5 and 15; the positions $t_1, t_2, \ldots, t_K$ of the change points are also random and such that $(t_1, t_2 - t_1, \ldots, T - t_K)/T$ follows a Dirichlet distribution with parameter $(\frac{2000}{K+1}, \frac{2000}{K+1}, \ldots) \in \mathbb{R}^{K+1}$; the mean shifts are random, such that on dimension i, the shift amplitude $|u_{t_{k-1},i} - u_{t_k,i}|$ follows a uniform distribution on $[0.9, 1.1]$. We add a Gaussian white noise with fixed variance on each of the ten dimensions. We report the average MSE and F-scores for different noise levels. Note that, here, the true number of changes is provided to the algorithms.

Initialization. For each signal, we initialize the set of centers $\{c_1, \ldots, c_V\}$ as follows. The signal is split into V equal-size segments, and the Fréchet mean is computed on each segment. The Fréchet means are the centers $\{c_1, \ldots, c_V\}$. We choose $V = 20$, which is larger than the expected number of changes.

Results. Looking at Figure 6, there is no clear difference between HOP and OP from the standpoint of the MSE and the F-score. Both metrics deteriorate when the noise level increases. This implies that HOP is close to OP, even though it only approximates the optimal solution to the change point detection problem.

E.3 Supplementary simulation: comparison between HOP and HMM

We compare HOP to a Gaussian HMM to do change point detection on noisy piecewise constant time series.

Data. We generate 400 signals as follows. They all have length $T = 10000$ and dimension $D = 8$. For a signal $(u_t)_{t=1}^{10000}$, a random number K of change point is uniformly chosen in $\{2, 4, 6, 8\}$; the positions $t_1, t_2, \ldots, t_K$ of

Table 1: Performance of HOP and Gaussian HMM on a synthetic data set (Section E.3)

	Precision (%)	Recall (%)	F-score (%)	Time
HOP	100 (±0)	100 (±0)	100 (±0)	17 ms (±7 ms)
HMM	19 (±38)	89 (±14)	19 (±38)	1.6 s (±0.2 s)

the change points are also random and such that $(t_1, t_2 - t_1, \dots, T - t_K)/T$ follows a Dirichlet distribution with parameter $(\frac{2000}{K+1}, \frac{2000}{K+1}, \dots) \in \mathbb{R}^{K+1}$; the mean shifts are random, such that on dimension i, the shift amplitude $|u_{t_{k-1},i} - u_{t_k,i}|$ follows a uniform distribution on $[1, 10]$. We add a Gaussian white noise with fixed variance $\sigma = 5$ on each of the eight dimensions. We report the average precision, recall, and F-score (with a margin of 1% of the signal's length).

Calibration. The true number of changes is provided to HOP. The number of states is set to $V = 20$. As for the Gaussian HMM, the number of states of set to K, the true number of changes. We use the implementation of HMMLearn[1].

Results. Results are shown in Table 1. On this data set, HOP has perfect scores, meaning that it can detect all change points. HMM has much lower precision: it detects too many change points. This is because HMM does not enforce a strong temporal persistence for the states, a fact which has been observed in other contexts (Nystrup et al., 2020). In terms of computation times, HOP is around 100 times faster than HMM on those signals.

E.4 Details on the synthetic experiments (Section 5)

Data set. The synthetic data set is generated as follows. Observations are 8×8 SPD matrices and follow a Wishart distribution with 10 degrees of freedom. Each time series has length $T \in \{1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000\}$ and $K^\star$ change points with $K^\star \in \{2, 4, 6, 8\}$. For each combination of $(T, K^\star)$, the positions $t_1, t_2, \dots, t_{K^\star}$ of the change points are random and such that $(t_1, t_2 - t_1, \dots, T - t_{K^\star})/T$ follows a Dirichlet distribution with parameter $(\frac{2000}{K^\star+1}, \frac{2000}{K^\star+1}, \dots) \in \mathbb{R}^{K^\star+1}$. On each segment, the observations follow a Wishart distribution with a fixed scale matrix. The mean is random and equal to $\boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^\top$ where $\boldsymbol{D}$ is a diagonal matrix whose diagonal elements follow a uniform distribution on $[0.1, 10]$ and $\boldsymbol{U}$ is an orthogonal matrix which follows the Haar distribution (uniform distribution on the orthogonal group).

Calibration. For the baseline algorithm OCP-Rie Wang et al. (2023), we use the authors' implementation, available at github.com/xiuheng-wang/CPD_manifold_release, with the default parameters. For gSeg we use the implementation available in the CRAN R package gSeg. For HOP, we initialize the states as in Section E.2.

E.5 Details on the human gait analysis with EMG signals (Section 6.1)

In Figure 7, we show the evolution of the F-score with respect to the margin. In Table 2, we show the F-scores for each walking type.

In Figure 8, we show the learned covariance matrices for a single signal (the same as in Figure 3).

E.6 Details on the mouse behavior analysis (Section 6.2)

Data. On this application, we applied our approach on a single video from Weinreb et al. (2024), namely open_field_2D/21_12_10_def6a_1_1.top.irDLC_resnet50_moseq_exampleAug21shuffle1_500000.mp4. The authors provided the positions of 8 2D markers (head, tail, spine, etc.) extracted with DeepLabCut (Mathis et al., 2018). To create an ethogram, we processed the time series of markers with Keypoint-Moseq using the default parameters from the Keypoint-Moseq tutorial[2] The tree representation of the ethogram was produced with the methodology and code of Luxem et al. (2022). Once the tree is computed, we use the library Networkx Hagberg et al. (2008) to compute the shortest path distances between nodes.

[1] github.com/hmmlearn/hmmlearn

[2] keypoint-moseq.readthedocs.io/en/latest/modeling.html

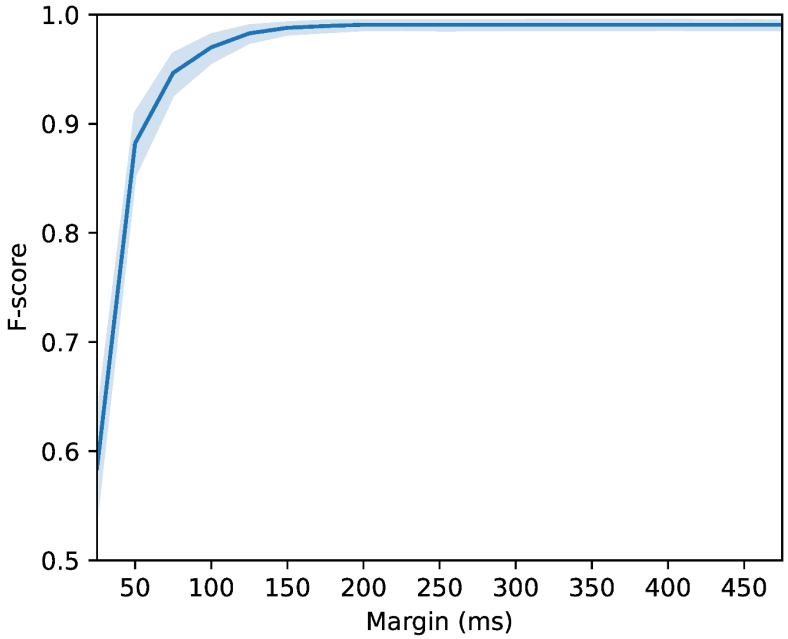


Figure 7: Evolution of the F-score with respect to the temporal margin (Section E.5)

Table 2: Performance on the EMG segmentation task, by activity type (margin: 75 ms)

Activity type	F-score	Precision	Recall
Backward walk	$97.7 \pm 4.3\%$	$97.5 \pm 4.6\%$	$98.0 \pm 4.1\%$
Heel walking	$81.7 \pm 22.0\%$	$81.2 \pm 24.0\%$	$82.7 \pm 19.7\%$
Incline walk	$98.8 \pm 7.1\%$	$98.8 \pm 7.1\%$	$98.8 \pm 7.1\%$
Normal walk	$94.3 \pm 15.4\%$	$93.3 \pm 16.5\%$	$95.4 \pm 14.6\%$
Toe walking	$80.6 \pm 22.3\%$	$79.5 \pm 22.8\%$	$81.8 \pm 22.1\%$
Walking butt kicks	$98.4 \pm 3.4\%$	$98.4 \pm 3.4\%$	$98.4 \pm 3.4\%$
Walking high knees	$95.3 \pm 13.1\%$	$95.6 \pm 13.2\%$	$95.1 \pm 13.3\%$

F Reproducibility

An implementation of the `HOP` algorithm is available at https://github.com/deepcharles/hop. Implemented in Python, it mainly uses Numpy Harris et al. (2020), Geomstats Miolane et al. (2020), and Numba Lam et al. (2015).

An archive containing data sets and notebooks to reproduce the experiments and generate the figures is available at https://drive.google.com/drive/folders/1gCV2R-Ry_SF1Vrv9cfTdeoMW3Jn9J2FM?usp=sharing (size: 2.4GB).

All results and timings have been executed on a Unix computer (Intel(R) Xeon(R) Gold 5220R CPU @ 2.20GHz) with 96 CPUs and 250 Gb of RAM.

References

A. A. Hagberg, D. A. Schult, and P. J. Swart. Exploring network structure, dynamics, and function using networkx. In *Proceedings of the Python in Science Conference*, 2008.

C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant. Array programming with numpy, 2020. ISSN 14764687.

S. K. Lam, A. Pitrou, and S. Seibert. Numba: A llvm-based python jit compiler. In *Proceedings of LLVM-HPC 2015: 2nd Workshop on the LLVM Compiler Infrastructure in HPC - Held in conjunction with SC 2015: The International Conference for High Performance Computing, Networking, Storage and Analysis*, LLVM '15, New York, NY, USA, 2015.

K. Luxem, P. Mocellin, F. Fuhrmann, J. Kürsch, S. R. Miller, J. J Palop, S. Remy, and P. Bauer. Identifying behavioral structure from deep variational embeddings of animal motion. *Communications Biology*, 5:1267, 2022.

A. Mathis, P. Mamidanna, K. M. Cury, T. Abe, V. N. Murthy, M. W. Mathis, and M. Bethge. Deeplabcut: markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience*, 21:1281–1289, 2018.

N. Miolane, N. Guigui, A. Le Brigant, J. Mathe, B. Hou, Y. Thanwerdas, S. Heyder, O. Peltre, N. Koep, H. Zaatiti, H. Hajri, Y. Cabanes, T. Gerald, P. Chauchat, C. Shewmake, D. Brooks, B. Kainz, C. Donnat, S. Holmes, and X. Pennec. Geomstats: A python package for riemannian geometry in machine learning. *Journal of Machine Learning Research (JMLR)*, 21, 2020.

Peter Nystrup, Erik Lindström, and Henrik Madsen. Learning hidden Markov models with persistent states by penalizing jumps. *Expert Systems with Applications*, 150:113307, 2020.

Karl-Theodor Sturm. Probability measures on metric spaces of nonpositive curvature. In *Heat Kernels and Analysis on Manifolds, Graphs, and Metric Spaces*, volume 338, pages 357–390. American Mathematical Society, Paris, France, contempora edition, 2002.

X. Wang, R. A. Borsoi, and C. Richard. Online Change Point Detection on Riemannian Manifolds With Karcher Mean Estimates. In *Proceedings of the IEEE European Signal Processing (EUSIPCO)*, pages 2033–2037, Helsinki, Finland, 2023.

Caleb Weinreb, Jonah E Pearl, Sherry Lin, Mohammed Abdal Monium Osman, Libby Zhang, Sidharth Annapragada, Eli Conlin, Red Hoffmann, Sofia Makowska, Winthrop F Gillis, et al. Keypoint-moseq: parsing behavior by linking point tracking to pose dynamics. *Nature Methods*, 21(7):1329–1339, 2024.

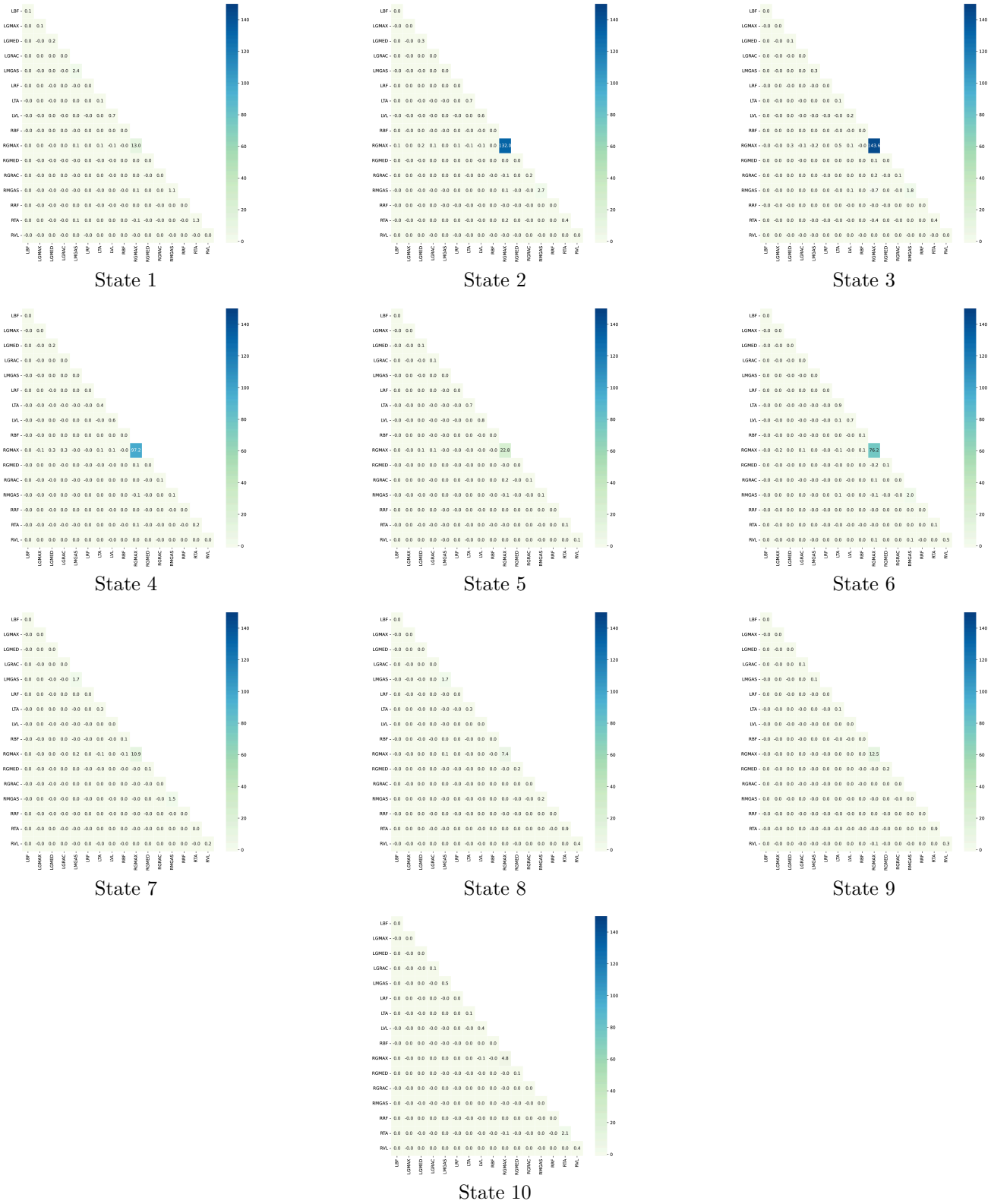


Figure 8: Covariance matrices associated with the states learned on the signal displayed on Figure 3