An Efficient Algorithm for Exact Segmentation of Large Compositional and Categorical Time Series

Charles Truong, Vincent Runge

Stat

WILEY

ORIGINAL ARTICLE OPEN ACCESS

An Efficient Algorithm for Exact Segmentation of Large Compositional and Categorical Time Series

Charles Truong[1] | Vincent Runge[2]

[1]Centre Borelli, Université Paris Saclay, Université Paris Cité, ENS Paris-Saclay, CNRS, SSA, INSERM, Gif-sur-Yvette, France | [2]Laboratoire de Mathématiques et Modélisation d'Evry, Université Paris-Saclay, CNRS, Univ Evry, Evry-Courcouronnes, France

Correspondence: Charles Truong (charles.truong@ens-paris-saclay.fr)

Received: 20 March 2024 | **Revised:** 19 June 2024 | **Accepted:** 5 September 2024

Keywords: change-point detection | compositional time series | dynamic programming | functional pruning

ABSTRACT

Change-point detection, also known as signal segmentation, is an essential preprocessing step in many applications, ranging from industrial monitoring to bioinformatics. In short, it consists in finding the temporal boundaries of homogeneous regimes in long and non-stationary time series. While this area of research is active, most existing methods are designed for Euclidean data. However, in many practical scenarios, the collected time series are compositional, meaning that each observation belongs to the probability simplex (the set of non-negative vectors whose components sum to one). In this work, we propose an algorithm detecting change-points in large compositional signals with an underlying piecewise stationary model. We cast the change-point detection task as a discrete optimization problem, whose solution is shown to converge to the true change-points. We introduce a new and time-efficient dynamic programming algorithm that solves exactly this problem. To limit the number of operations, we describe a novel pruning rule that allows us to reduce the set of candidate change-point indices. Our method is tested on a thorough simulation study, which confirms its efficiency. Additionally, we apply our method to a human activity segmentation task, highlighting the necessity for such novel techniques compared to standard algorithms.

1 | Introduction

For nearly a century, researchers have been exploring the task of detecting changes in the underlying model of time series, known as change-point detection or signal segmentation (Tartakovsky, Nikiforov, and Basseville 2014). It consists of partitioning a signal into contiguous segments, for which all data in each segment should share the same underlying statistical structure. This field has not only generated significant interest in statistics, signal processing and data mining community, but it has also found numerous practical applications, including speech processing (Seichepine et al. 2014), financial analysis (Ueda, Ike, and Yamanishi 2022) and bioinformatics (Truong, Oudre, and Vayatis 2020). For instance, medical researchers might be interested in detecting changes in the activity of subjects they have monitored over long periods of time (Jung et al. 2021), enabling them to compute meaningful statistics on their patients' physical states. In this context, change-point detection is the task of finding the temporal boundaries of each activity. In cyber-security, detecting the onset of specific attacks, such as denial-of-service attacks, can also be framed as a change-point detection problem (Carl et al. 2006; Lung-Yut-Fong, Lévy-Leduc, and Cappé 2011). Most of the existing literature on this subject is dedicated to signals that take values in an Euclidean space. In addition, current exact methods tend to scale poorly with the number of observations and are often replaced by approximations when changes are more complex than Gaussian mean shift and signals have more than several thousand samples (Truong, Oudre, and Vayatis 2020). Recently, researchers have attempted to broaden the applicability of such methods to larger time series with non-numeric or structured data such as text (Cohen, Heeringa, and Adams 2002), ordinal data (Lung-Yut-Fong, Lévy-Leduc, and Cappé 2011), network data (Barnett and Onnela 2016) and so forth.

In this study, our focus is on compositional time series. Compositional time series are sequences where each element is a vector of proportions or probabilities that sum to 100% or 1. Examples of compositional time series include market shares of products, the percentage distribution of different types of land use in a region or the allocation of expenses in a budget over time. Analysing compositional signals requires specialized statistical techniques that take into account the shape of the space in which observations lie. Several works have extended classical signal processing and statistical procedures (Rieser and Filzmoser 2023; Pawlowsky-Glahn and Egozcue 2016). Also, methods such as log-ratio transformations are often employed to map the data to the Euclidean space (Godichon-Baggioni, Maugis-Rabusseau, and Rau 2019). However, they are not well defined for data with components equal to 0. Therefore, approaches that take into account the geometry of the compositional space without transformation are needed. For the change-point detection task, current methods only focus on detecting a change in the parameter of a Dirichlet distribution (Prabuchandran et al. 2022; Fisher et al. 2022).

Categorical signals, which are sequences of observations belonging to a finite set (an alphabet) of symbols, can be seen as an edge case of compositional signals. Indeed, using the one-hot encoding scheme, each symbol can be mapped to a binary compositional vector that is zero everywhere except at one position, where it is equal to one. Categorical data are prevalent in various applications, dealing with DNA sequences, computer logs or web pages (Jian et al. 2019), for instance. In addition, in an increasing number of works, numerical time series are transformed into categorical signals either to decrease the memory and computational load (Li and Lin 2017) or to summarize long and complex data in an interpretable manner (Germain et al. 2023). This process is often referred to as discretization or symbolization. Consequently, classical procedures such as clustering (Kelil and Wang 2008), anomaly detection (Wu and Wang 2013), and pattern discovery (Alaee et al. 2021) are being adapted to handle categorical signals; see García et al. (2013) for a survey. Note that the setting of categorical signals is slightly different from the setting of Wang, Zou, and Yin (2018), where observations follow a multinomial (and not categorical) distribution with piecewise constant parameters.

Change-point detection for compositional and categorical data is challenging as most algorithms are designed for vector-valued data. The present work is in line with the current trend and aims to provide an efficient change-point detection algorithm that can cope with the specific nature of the signals and their sizes.

Our contributions are threefold:

- We define the problem of change-point detection for categorical and compositional time series and describe the optimization problem that leads to the change-point estimators.
- The estimators are shown to be consistent with the true change-points.
- Based on a new pruning rule, we provide an efficient algorithm for large signals.

Section 2 presents the compositional multiple change-point problem and reformulates it as an optimization problem. Section 3 introduces our new algorithm for change-point detection. In a simulation study in Section 4, we highlight the efficiency of the proposed method, followed by an application to human activity sensor data in Section 5.

2 | A Model for Non-Stationary Compositional Signals

2.1 | Model

Denote by $\boldsymbol{y}_{0..n} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)$ a signal of n observations taking values in the probability simplex $\Delta^{D-1} := \left\{ \boldsymbol{q} \in \mathbb{R}^D \text{ s.t. } 0 \leq q_i \leq 1 \text{ and } \sum_i q_i = 1 \right\}$. Assume that there are $K^\star$ change-points located at the indices $t_1^\star, \ldots, t_{K^\star}^\star$, meaning that on each segment between $t_k^\star + 1$ and $t_{k+1}^\star$, the expected value of $\boldsymbol{y}_t$ is constant:

$$\mathbb{E}\boldsymbol{y}_t = \boldsymbol{p}^{\star(k+1)} \text{ if } t_k^\star < t \leq t_{k+1}^\star, \tag{1}$$

where $\boldsymbol{p}^{\star(k+1)} \in \Delta^{D-1}$ are the true but unknown parameter values with $\boldsymbol{p}^{\star(k)} \neq \boldsymbol{p}^{\star(k+1)}$. Here, only the first moment of the distribution is assumed to be piecewise constant. The higher order moments can vary with time. This includes the well-known signal-plus-noise model where the signal is piecewise constant (and each element belongs to the simplex), but contrary to many settings, we do not assume that the noise is i.i.d. We aim at recovering the number of changes $K^\star$ and the change-point indices $t_1^\star, \ldots, t_{K^\star}^\star$.

Note that this setting includes categorical time series as a limit case. Indeed, categorical time series take values in a finite set of symbols. By using the one-encoding scheme, we can convert any categorical sample to a vector in Δ^{D-1} whose components are zero everywhere except at the kth position, where it is equal to one if the sample is equal to the kth symbol of the set. This is the case, somehow, of the largest possible variance, as all observations are the farthest possible from the underlying mean vector in the simplex. Here, a shift in $\mathbb{E}\boldsymbol{y}_t$ is equivalent to a change in the parameter of a multinomial distribution.

2.2 | Proposed Estimator

Our estimation of the change-point indices is based on the maximization of the likelihood of Model 1. To formally introduce the estimator, define the quantities $c(\boldsymbol{y}_{a..b}, \boldsymbol{p})$ and $c(\boldsymbol{y}_{a..b})$ for $\boldsymbol{p} \in \Delta^{D-1}$ and $\boldsymbol{y}_{a..b} := (\boldsymbol{y}_{a+1}, \ldots, \boldsymbol{y}_b)$ the sub-signal between indices $a+1$ and b as follows:

$$c(\boldsymbol{y}_{a..b}, \boldsymbol{p}) := -\sum_{t=a+1}^{b} \boldsymbol{y}_t^\top \log \boldsymbol{p} \text{ and } c(\boldsymbol{y}_{a..b}) := \min_{\boldsymbol{p} \in \Delta^{D-1}} c(\boldsymbol{y}_{a..b}, \boldsymbol{p}). \tag{2}$$

By convention, in the following, $0 \log 0 = 0$. The function $c(\cdot)$ is referred to as the (segment) cost function in the change-point detection literature (Killick, Fearnhead, and Eckley 2012). We now define our estimator of the change-point number and locations.

Definition 1. The estimators of the number of change-points and their locations, denoted $\widehat{K}$ and $\widehat{t}_1, \ldots, \widehat{t}_K$, are defined by

$$\widehat{K}, \widehat{t}_1, \ldots, \widehat{t}_K := \underset{K, 0 \leq t_1 < t_2, \ldots < t_K < n}{\operatorname{argmin}} \left[\sum_{k=0}^{K} c(\boldsymbol{y}_{t_k..t_{k+1}}) + \beta K \right], \quad (3)$$

where $\beta > 0$ is a user-defined parameter.

The parameter β in (3) is a penalty that controls the trade-off between the data fidelity term (sum of segment costs) and the model complexity (measured by the number K of changes). A large β detects few segments, and a small β produces many segments. Finding an appropriate β is a model selection task. Model selection for change-point detection is an active research subject. In parametric settings, optimal penalty values have known shape, e.g. for data that are piecewise Gaussian (Lebarbier 2005; Verzelen et al. 2020; Zhang and David 2007) or from a distribution of the exponential family (Cleynen and Lebarbier 2017). However, our setting does not assume a parametric distribution for the noise process. Data-driven heuristics, such as the slope heuristics (Arlot 2019), cross-validation for change-point detection (Pein and Shah 2021; Chen et al. 2024) or sample splitting (Zou, Wang, and Li 2020), can still be applied here, even though their theoretical guarantees will not be valid in our context. In certain applications, the number and locations of changes are known for a few signals (the training set). One could then use supervised approaches, such as cross-validation (as is done in the simulation study) or more involved learning strategies (Hocking et al. 2020; Blotas and Truong 2024).

Thus defined, it is not clear why the estimators $\widehat{K}$ and $(\widehat{t}_k)_k$ would converge to the true $K^\star$ and $(t_k^\star)_k$. This will be proven rigorously in the next section. However, in the case of categorical time series, the cost function $c(\cdot)$ has a simple interpretation, explained in the following remark.

Remark 1. Denote by $\boldsymbol{x}_{0..n} = (x_1, \ldots, x_n)$ a signal of n independent observations taking values in a finite set (or alphabet) $\mathcal{U} = \{u_1, \ldots, u_D\}$ of size D. Each data point x_t is a multinomial random variable whose parameter is $\boldsymbol{p}_t$; that is, $p_{t,i} := P(x_t = u_i)$. Further define the one-hot encoded vector $\boldsymbol{y}_t \in \Delta^{D-1}$ by $y_{t,d} = \mathbf{1}(x_t = u_d)$. Straightforwardly, $\boldsymbol{p}_t = \mathbb{E}\boldsymbol{y}_t$. Then, $c(\boldsymbol{y}_{a..b}, \boldsymbol{p})$ is the negative log-likelihood on the subsignal $\boldsymbol{y}_{a..b}$, and if $\boldsymbol{p}_t$ is assumed to be piecewise constant, the estimators $\widehat{K}$ and $\widehat{t}_k$ maximize a penalized log-likelihood. In that context, a possible value for β is derived from the Bayesian information criterion (BIC) (Yao 1988): $\beta_{\text{BIC}} := (D-1)\log(n)/2$.

2.3 | Theoretical Study

The estimators (3) belong to a larger class of estimators that has been studied from a theoretical standpoint (Lavielle 1999). In the following, we apply a result from (Lavielle 1999) to our setting to prove that the estimated change-point number and locations are asymptotically consistent when the number n of samples grows to infinity and for a well-chosen β. Remarkably, this consistency result holds even if the observations are dependent, which is often the case in practice. First, we state an assumption on the 'noise' process, that is, the deviation from the average behaviour.

Assumption 1. For any $\boldsymbol{p} \in \Delta^{D-1}$, define the process $z_t(\boldsymbol{p}) := -\left[\boldsymbol{y}_t - \mathbb{E}\boldsymbol{y}_t\right]^\top \log \boldsymbol{p}$. There exists $1 \leq h < 2$ and a constant $C(\boldsymbol{p})$ that depends on $\boldsymbol{p}$ only such that

$$\mathbb{E}\left[\left(\sum_{t=a+1}^{b} z_t(\boldsymbol{p})\right)^2\right] \leq C(\boldsymbol{p})(b-a)^h \text{ for any } 0 \leq a < b \leq n. \quad (4)$$

Assumption 1 is verified by several process models. For instance, if $\boldsymbol{y}_{0..n}$ is a sequence of independent random variables, then (4) holds with $h = 1$. Also, if $z_t(\boldsymbol{p})$ is second-order stationary with covariance function $\gamma_{z,\boldsymbol{p}}(\cdot)$ such that $\gamma_{z,\boldsymbol{p}}(k) = \mathcal{O}(k^{-l})$ for a $l > 0$, then Assumption 1 is satisfied with $h = \max(2-l, 1)$. See Lavielle (1999) for a discussion about processes for which Assumption 1 holds.

We are now ready to state the main theoretical result. The proof is deferred to the appendix.

Theorem 1. Let $(\beta_n)_n$ be a sequence of positive numbers such that $\beta_n/n \to 0$ and $n^{1-h}\beta_n \to \infty$ as n grows to infinity. Then, under Assumption 1, the estimated number of changes $\widehat{K}$ and the estimated (normalized) change-point locations $\widehat{t}_k/n$ defined in (3) with $\beta = \beta_n$ converge in probability to $K^\star$ and $\tau_k^\star := t_k^\star/n$ respectively. Formally,

$$\begin{aligned} &\lim_{n\to\infty} \mathbb{P}\left(\widehat{K} \neq K^\star\right) = 0 \text{ and } \forall \epsilon > 0, \\ &\lim_{n\to\infty} \mathbb{P}\left(\max_{k=1,\ldots,K^\star} |\widehat{t}_k/n - \tau_k^\star| \geq \epsilon\right) = 0. \end{aligned} \quad (5)$$

3 | Optimization For Large Compositional Signals

To compute the estimators from Definition 1, one has to solve a complex discrete optimization problem. A straightforward approach is to enumerate all possible segmentations $(K, t_1, \ldots, t_K)$ and return the one that minimizes the objective function (3). However, there are 2^{n-1} possible segmentations; therefore, this approach is not tractable even for small signals. Methods based on dynamic programming have been developed to solve such optimization problems. This section describes current approaches designed to cope with large compositional signals and our method.

3.1 | Related Work on Dynamic Programming Approaches

Dynamic programming algorithms for change-point detection are based on the objective function's additive nature (over segments) in (3). Indeed, this allows us to compute the optimum recursively. To make this statement precise, introduce V_t, the optimal sum of costs over partial data $\boldsymbol{y}_{0..t}$:

$$V_t := \min_{K, 0 \leq t_1 < \ldots < t_K < t} \left[\sum_{k=0}^{K} c(\boldsymbol{y}_{t_k..t_{k+1}}) + \beta K \right]. \quad (6)$$

The quantity V_t corresponds to the (penalized) negative maximum log-likelihood for the observations from 1 to t. Dynamic programming is based on the following recursion:

$$V_t = \min_{s=0,\dots,t-1} \left[V_s + c(\boldsymbol{y}_{s..t}) + \beta \right], \tag{7}$$

with $V_0 = 0$. Thanks to (7), it is possible to compute $V_1, V_2, \dots, V_n$ recursively. By saving the best last change-point location — the argminimum in (7) — for all t from 1 to n, we can recover the minimizer of (3). This algorithm, first described in Jackson et al. (2005) and called optimal partitioning (OP), has a time complexity of order $\mathcal{O}(n^2)$.

A quadratic complexity can be cumbersome for applications with long signals. As a result, methods to decrease the number of operations have been developed. A well-known approach is called `PELT` (Pruned Exact Linear Time) (Killick, Fearnhead, and Eckley 2012) and has a complexity of the order of $\mathcal{O}(n)$ under certain assumptions over the data. To put it briefly, the objective of `PELT` is to find a set of indices $\mathcal{A}_t \subseteq \{0,1, \dots, t-1\}$ such that the recursive relation (7) still holds:

$$V_t = \min_{s \in \mathcal{A}_t} \left[V_s + c(\boldsymbol{y}_{s..t}) + \beta \right]. \tag{8}$$

The complexity of computing V_t from the V_s $(s < t)$ then decreases from $\mathcal{O}(t)$ to $\mathcal{O}(|\mathcal{A}_t|)$, provided that computing the cost $c(\boldsymbol{y}_{s..t})$ has constant complexity. Ideally, the size of $\mathcal{A}_t$ is much smaller than t. In Killick, Fearnhead, and Eckley (2012), the authors show that the following simple inequality test can prune many indices: if, for $s < t$,

$$V_s + c(\boldsymbol{y}_{s..t}) > V_t, \tag{9}$$

then s can never be the last change-point of a segmentation of $\boldsymbol{y}_{0..T}$ for any $T \geq t$. In other words, by setting $\mathcal{A}_t = \{0,1, \dots, t-1\} \setminus \{s \text{ s.t. } (9) \text{ holds}\}$, the recursive relation (8) is satisfied and one can compute the V_t progressively but with less operations.

3.2 | DuST: A New Simple Pruning Method

This section describes a new pruning rule, in the spirit of `PELT`'s pruning rule (9). We aim to reduce the size of $\mathcal{A}_t$ and, therefore, the total complexity.

Before introducing our new pruning rule, let us introduce some further notations. Define $V_t^s(\theta)$, the cost of the optimal segmentation of $\boldsymbol{y}_{0..t}$ whose last change-point is $s < t$ and the probability vector on the last segment is $\boldsymbol{p} = \exp\boldsymbol{\theta}$:

$$V_t^s(\boldsymbol{\theta}) := V_s + c(\boldsymbol{y}_{s..t}, \boldsymbol{\theta}) + \beta. \tag{10}$$

Similarly, V_t^s is the cost of the optimal segmentation of $\boldsymbol{y}_{0..t}$ whose last change-point is $s < t$:

$$V_t^s := \min_{\boldsymbol{\theta} \in \Theta} V_t^s(\boldsymbol{\theta}) = V_s + c(\boldsymbol{y}_{s..t}) + \beta. \tag{11}$$

Finally, let $V_t(\boldsymbol{\theta})$ be the cost of the optimal segmentation of $\boldsymbol{y}_{0..t}$ whose probability vector on the last segment is $\boldsymbol{p} = \exp\boldsymbol{\theta}$:

$$V_t(\boldsymbol{\theta}) := \min_{s=0,\dots,t-1} \left[V_s + c(\boldsymbol{y}_{s..t}, \boldsymbol{\theta}) + \beta \right] = \min_{s=0,\dots,t-1} \{ V_t^s(\boldsymbol{\theta}) \}. \tag{12}$$

Because of their dependence on parameter $\boldsymbol{\theta}$, the quantities $V_t(\boldsymbol{\theta})$ and $V_t^s(\boldsymbol{\theta})$ are regarded as the functional counterparts of V_t and V_t^s (Maidstone et al. 2017).

We can now state a first pruning rule based on those functional quantities. This rule is ideal (in a sense that will be described later) but computationally cumbersome. It will serve as a basis for our actual rule.

Proposition 1. Functional pruning rule Define $\Theta_t^s \subset \Theta$ and m_t^s by

$$\Theta_t^s := \{ \boldsymbol{\theta} \in \Theta \mid \forall s' \neq s, V_t^s(\boldsymbol{\theta}) \leq V_t^{s'}(\boldsymbol{\theta}) \} \tag{13}$$

and

$$m_t^s := \min_{\boldsymbol{\theta} \in \Theta_t^s} V_t^s(\boldsymbol{\theta}). \tag{14}$$

By convention, if $\Theta_t^s = \emptyset$, then $m_t^s = +\infty$. The functional pruning condition is as follows: if, for $s < t$,

$$m_t^s > V_t + \beta, \tag{15}$$

then s can never be the last change-point of a segmentation of $\boldsymbol{y}_{0..T}$ for $T \geq t$.

Proof. Let $\boldsymbol{\theta} \in \Theta_t^s$. If the functional pruning condition (15) holds, we have

$$V_T^s(\boldsymbol{\theta}) = V_t^s(\boldsymbol{\theta}) + c(\boldsymbol{y}_{t..T}, \boldsymbol{\theta}) > V_t + \beta + c(\boldsymbol{y}_{t..T}, \boldsymbol{\theta}) = V_T^t(\boldsymbol{\theta}) \geq V_T.$$

If $\boldsymbol{\theta} \notin \Theta_t^s$, there exists s' such that $V_t^s(\boldsymbol{\theta}) > V_t^{s'}(\boldsymbol{\theta})$ and, therefore,

$$V_T^s(\boldsymbol{\theta}) = V_t^s(\boldsymbol{\theta}) + c(\boldsymbol{y}_{t..T}, \boldsymbol{\theta}) > V_t^{s'}(\boldsymbol{\theta}) + c(\boldsymbol{y}_{t..T}, \boldsymbol{\theta}) = V_T^{s'}(\boldsymbol{\theta}) \geq V_T.$$

As a result, for all $\boldsymbol{\theta} \in \Theta$, $V_T^s(\boldsymbol{\theta}) > V_T$, and the index s can be discarded for all $T \geq t$. □ □

Thanks to Proposition 1, by setting $\mathcal{A}_t = \{0,1, \dots, t-1\} \setminus \{s \text{ s.t. } (9) \text{ or } (15) \text{ holds}\}$, the recursive relation (8) remains true and one can still use dynamic programming to find change-points. Using an approach similar to Pishchagina, Rigaill, and Runge (2023), it is possible to show that the functional pruning rule is maximal, meaning that $\{s \text{ s.t. } (15) \text{ holds}\}$ is the largest set of indices that can be pruned while keeping the recursive relation (8) satisfied for any signal $y_{0..n}$. Unfortunately, computing the value of m_t^s (14) requires solving a non-convex minimization problem for each $t = 1, \dots, n$, which is computationally heavy. Even for Gaussian models, it cannot be effectively tackled (Runge 2020), except for small dimensions by using geometric approximations of the optimization problem (Pishchagina, Rigaill, and Runge 2023).

Our strategy to cope with this issue is to find an easy-to-compute lower bound of the quantity m_t^s (14). Indeed, if this

lower bound is larger than $V_t + \beta$, then the functional pruning (15) is satisfied, and the index s can be pruned. To that end, define for a triplet (s', s, t) of indices such that $s' < s < t$ the quantity $m_t^{s',s}$ by

$$m_t^{s',s} := \min_{\theta} V_t^s(\theta) \text{ s.t. } \sum_{i=1}^{d} e^{\theta_i} = 1 \text{ and } V_t^s(\theta) \leq V_t^{s'}(\theta). \quad (16)$$

It is straightforward to see that $m_t^{s',s} \leq m_t^s$ since it is the minimum of the same objective function over a larger space (only one constraint of Θ_t^s (13) is considered). However, the optimization problem (16) remains non-convex and difficult to solve. A simpler lower bound can be obtained from the Lagrange dual function

$$\mathcal{L}(\mu, \lambda) := \min_{\theta \in \mathbb{R}^d} \left[V_t^s(\theta) + \mu\left(V_t^s(\theta) - V_t^{s'}(\theta)\right) + \lambda\left(e^{\theta_i} - 1\right) \right], \quad (17)$$

where $\lambda \in \mathbb{R}$ and $\mu \geq 0$ are the so-called Lagrange multipliers. A well-known result in optimization theory states that the Lagrange dual function is a lower bound on the optimal value $m_t^{s',s}$ of problem (16) (Boyd and Vandenberghe 2004), that is, $\mathcal{L}(\mu, \lambda) \leq m_t^{s',s}$. Consequently, $\mathcal{L}(\mu, \lambda) \leq m_t^s$. Combining this observation with the functional pruning rule of Proposition 1 yields our suggested pruning condition: At iteration t, prune s if there exists s' such that $\mathcal{L}(\mu, \lambda) > V_t + \beta$ for certain $\lambda \in \mathbb{R}$ and $\mu \geq 0$. The following proposition is a precise formulation of this condition, with a carefully chosen pair (λ, μ) and several algebraic simplifications. Our new pruning rule is named DuST for duality sample test.

Proposition 2. DuST's pruning rule Let (s', s, t) denote a triplet of indices such that $s' < s < t$ and define

$$\overline{\mu} := \min_{i=1,\dots,d} \{ S_{s'..s,i} / S_{s..t,i} \,|\, S_{s'..s,i} > 0 \}, \quad (18)$$

where the $\mathbf{S}_{a..b} := \boldsymbol{y}_{a+1} + \dots + \boldsymbol{y}_b$ are the partial sums of $\boldsymbol{y}_{0..n}$. If there exists $\mu \in (0, \overline{\mu})$ such that

$$H(\boldsymbol{p}(\mu)) > \frac{V_t - V_s - \mu(V_s - V_{s'})}{(t-s) - \mu(s-s')}, \quad (19)$$

where $H(\boldsymbol{p}) := -\sum_i p_i \log p_i$ is the Shannon entropy of the discrete probability $\boldsymbol{p}$ and

$$\boldsymbol{p}(\mu) := \frac{\mathbf{S}_{s..t} - \mu \mathbf{S}_{s'..s}}{(t-s) - \mu(s-s')}, \quad (20)$$

then s can never be the last change-point of a segmentation of $\boldsymbol{y}_{0..T}$ for $T \geq t$.

The proof is deferred to the appendix. Interestingly, when setting $\mu = 0$ in inequality (19), the pruning rule is exactly PELT's condition (9). In that regard, DuST can be seen as an extension of PELT.

The procedure that uses DuST is described in Algorithm 1. At iteration t, possible change-point positions are saved into the set $\mathcal{A}_t \subset \{0, \dots, t-1\}$. In addition to PELT test, for each $s \in \mathcal{A}_t$, we perform two draws: choose uniformly at random an index $s' \in \mathcal{A}_t$ with $s' < s$ and choose μ uniformly on $(0, \overline{\mu})$ where $\overline{\mu}$ depends on indices s' and s and current time t. Then if condition 19 holds, the index s is discarded. The space complexity of DuST is linear in the number of samples. The worst-case time complexity of this algorithm remains of the order of $\mathcal{O}(n^2)$ as in PELT. However, since DuST is guaranteed to prune more indices than PELT, it is reasonable to assume that it enjoys at least similar average-case complexity, which is linear in n under certain conditions (Killick, Fearnhead, and Eckley 2012). On simulations, DuST test is capable of pruning within segment indices while PELT test struggles to do so. Theoretical analysis of the complexity of DuST is left for future work.

4 | Simulation Study

This section describes the performance of DuST on simulated categorical data for a varying signal length.

4.1 | Simulation Setting

4.1.1 | Baseline Methods

First, our algorithm is compared to a standard change-in-mean change-point detection procedure in order to assess detection accuracy. This method, denoted MS-CPD for Mean Shift Change-Point Detection (Truong, Oudre, and Vayatis 2020), finds the best piecewise constant approximation of the signal $\boldsymbol{y}_{0..n}$. The underlying model considers the signal as Gaussian with constant variance and independence across dimensions. Similarly to our method DuST, a penalty β controls the number of changes. Second, our algorithm DuST is compared to the PELT algorithm to understand the computational gain of our pruning rule. Note that both DuST and PELT outputs the exact same segmentation. In addition, the code of our method DuST is available online.[1]

4.1.2 | Evaluation Metrics

To assess the accuracy of change-point detection methods, we use two metrics: the adjusted rand index (ARI) and the F1-score. The ARI is a well-known metric (Hubert and Arabie 1985) when comparing clustering algorithms (Warrens and van der Hoef 2022) and is broadly used in change-point analysis (Fearnhead and Rigaill 2020; James and Matteson 2013; Londschien, Kovács, and Bühlmann 2021). Roughly, the ARI between two sets of change-points $\mathcal{T}^\star = \{t_1^\star, t_2^\star, \dots\}$ (true change-points) and $\widehat{\mathcal{T}} = \{\widehat{t}_1, \widehat{t}_2, \dots\}$ (estimated change-points) counts the number of agreements between $\mathcal{T}^\star$ and $\widehat{\mathcal{T}}$. An agreement is a pair of indices (s, t), which are either in the same segment according to both $\mathcal{T}^\star$ and $\widehat{\mathcal{T}}$ or in different segments according to both $\mathcal{T}^\star$ and $\widehat{\mathcal{T}}$. The number of agreements is then adjusted as in Sundqvist, Chiquet, and Rigaill (2022): It is 1 if $\mathcal{T}^\star$ and $\widehat{\mathcal{T}}$ are equal and has an expected value of 0 if $\mathcal{T}^\star$ and $\widehat{\mathcal{T}}$ are independent.

The F1-score is defined as follows. A true change-point $t_k^\star$ is declared 'detected' if at least one estimated change within a user-defined margin exists. Precision and recall are given by

Algorithm 1 DuST algorithm.

Require: Compositional signal $\mathbf{y}_{0..n}$, penalty value $\beta > 0$

Ensure: Set of change-point indexes $\widehat{\mathcal{T}} = \{\hat{t}_1, \hat{t}_2, \dots\}$

$V_0 \leftarrow 0, \quad \mathcal{A}_1 \leftarrow \{0\}$

for $t = 1, \dots, n$ **do** ▷ Forward pass

  $V_t \leftarrow \min_{s \in \mathcal{A}_t} \left[V_s + c(\mathbf{y}_{s..t}) + \beta\right]$

  $\hat{\tau}_t \leftarrow \arg\min_{s \in \mathcal{A}_t} \left[V_s + c(\mathbf{y}_{s..t}) + \beta\right]$

  for $s \in \mathcal{A}_t$ **do**

    if Condition (9) holds **then** ▷ PELT pruning

      $\mathcal{A}_t \leftarrow \mathcal{A}_t \backslash \{s\}$

    else

      Draw $s' \in \mathcal{A}_t$ uniformly such that $s' < s$.

      Compute $\bar{\mu}$ (18) and draw μ uniformly from $(0, \bar{\mu})$.

      if Condition (19) holds **then** ▷ DuST pruning

        $\mathcal{A}_t \leftarrow \mathcal{A}_t \backslash \{s\}$

      end if

    end if

  end for

  $\mathcal{A}_{t+1} \leftarrow \mathcal{A}_t \cup \{t\}$

end for

$t \leftarrow n, \ \widehat{\mathcal{T}} \leftarrow \emptyset$

while $t > 0$ **do** ▷ Backtracking

  $\widehat{\mathcal{T}} \leftarrow \widehat{\mathcal{T}} \cup \{t\}$

  $t \leftarrow \hat{\tau}_t$

end while

precision: $= |\text{TP}|/\widehat{K}$ and recall: $= |\text{TP}|/K^\star$ where, for a given margin M, the true positives TP are true changes for which there is an estimated one at less than M samples, that is, $\text{TP}: = \{t_k^\star \,|\, \exists \widehat{t}_l \text{ s.t. } |\widehat{t}_l - t_k^\star| \leq M\}$. We also add the constraint that a single estimated event cannot detect two true events. The F1-score for a margin M, denoted F1@M, is the geometric mean of precision and recall.

To measure the efficiency of pruning rules, we also report the percentage of non-pruned indices, that is, $\sum |\mathcal{A}_t|$ in Algorithm 1 divided by $n(n-1)/2$, the number of non-pruned indices in the worst-case (absence of pruning). A low percentage of non-pruned indices means that few operations are needed to detect change-points.

4.1.3 | Simulated Data

We generate 100 categorical signals with a fixed length of 50,000 samples, a random number of segments ranging from 8 to 12 and a fixed number of symbols ($d = 5$). The segmentation is uniform, meaning that all segments have the same length. The distribution of symbols alternates between segments: On odd segments, the probability vector is (0. 02,0. 07,0. 16,0. 29,0.45), and on even segments, (0. 07,0. 02,0. 16,0. 29,0.45) (the first two coefficients are swapped). For both `DuST` and `MS-CPD`, the value of β is calibrated with a grid search going from 10^{-1} to 10^1 on a separate set of 10 signals. The chosen value of β is the one in the grid that minimizes the error on the number of changes $|K^\star - \widehat{K}|$. This approach is often used when users have already labelled a few training time series (Hocking et al. 2020; Blotas and Truong 2024). For all metrics, the average and standard deviation are reported. All experiments ran on a 2.20GHz Intel(R) Xeon(R) Gold 5220R CPU with 252 Gb RAM.

4.1.4 | Results

The segmentation accuracy is shown on Table B1 and Figure B1. Several comments can be made:

TABLE B1 | Results on the simulated data set.

Metric	DuST	MS-CPD
ARI	**0.99** (± **0.00**)	0.98 (± 0.02)
$\lvert\widehat{K} - K^\star\rvert$	**0.0** (± **0.0**)	0.4 (± 0.6)
F1@100 samples	**97.8%** (± **5.0%**)	89.7% (± 9.9%)
Prec@100 samples	**97.8%** ( )	88.2% (± 11.1%)
Rec@100 samples	**97.8%** (± **5.0%**)	91.3% (± 9.0%)

Note: Here, $K^\star$ is the true number of change-points, and $\widehat{K}$ is the estimated one. For ARI and F1-score, higher is better; for $|\widehat{K} - K^\star|$, lower is better. The best score is in bold.

- According to all metrics displayed on Table B1, DuST performs better than MS-CPD. Our method can correctly estimate the number of changes ($|\widehat{K} - K^\star|$ is zero) and localize them (F1-score close to 100%). As shown in Figure B1, this conclusion holds uniformly for all margins, meaning that even with a more conservative F1-score or a more tolerant F1-score, DuST remains better.
- MS-CPD does not always predict the correct number of changes. Also, the detected changes are not well localized, as evidenced by the low precision (Prec@100 samples) compared to DuST.
- Both these observations indicate that treating a categorical signal as an Euclidean signal is sub-optimal for the change-point detection task.
- In addition, for signals close to our proposed Model 1, calibrating the penalty value β on a small training set yields good performance. By comparison, a single penalty value for MS-CPD cannot predict the correct number of changes for similar signals.

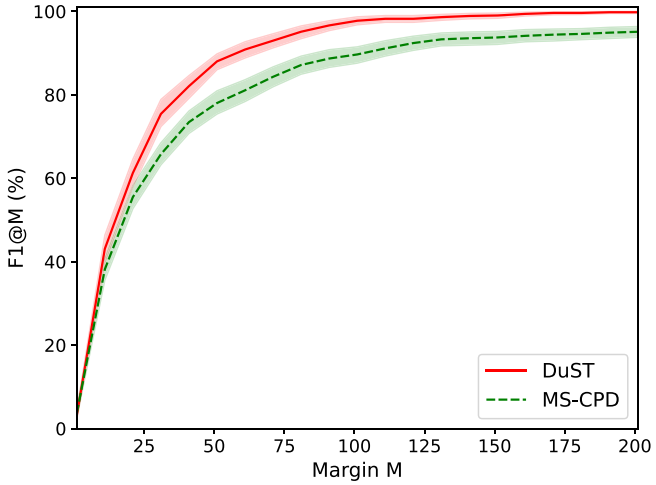


FIGURE B1 | Evolution of the F1-score w.r.t. the margin on the simulated data.

From a computational standpoint, DuST and PELT are compared, and the results are displayed in Figure B3. Recall that both methods output the exact same change-points. The only difference is that DuST has an additional pruning rule. We can draw several observations from the results:

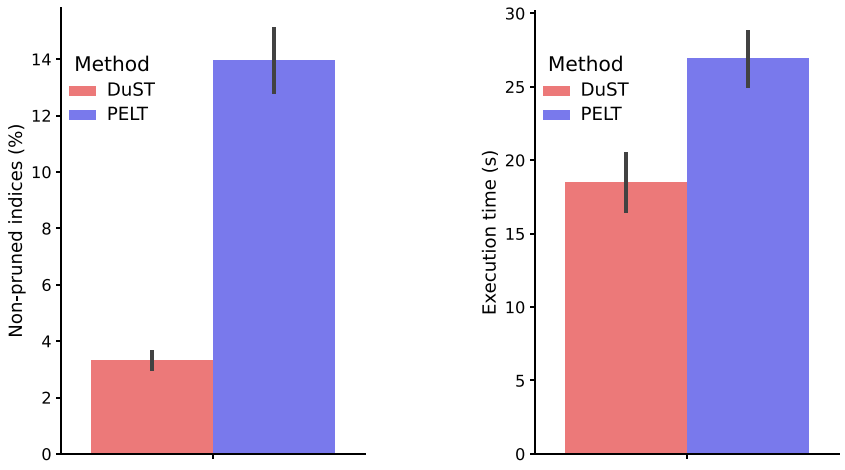
- As expected, DuST prunes more than PELT. On the simulated data set, DuST keeps less than three times indices (Figure B2a). This translates into a lower execution time (Figure B2b). However, the gain is not as important in terms of time because of the overhead of evaluating if DuST pruning rule (19) is satisfied.
- When the number of samples increases, we empirically see that PELT remains around 10% of non-pruned indices while DuST reaches almost 1% (Figure B3c). As for the execution time, the difference between the two algorithms increases as n grows. This means that for larger and larger signals, DuST is more and more advantageous (Figure B3d). This is in line with the observation that DuST can prune indices even for time series with a small ratio number of change over data length, contrary to PELT.

When there are many changes, it has been observed in Pishchagina, Rigaill, and Runge (2023) (for a Gaussian model) that the computational overhead of a functional pruning rule is larger than the gain, leading to slightly longer execution times for small to medium n (in the thousands). It can be expected that DuST behaves similarly. As a solution, we can check DuST's rule after a given delay when $t - s$ is above a user-defined threshold, which would lead to a trade-off between pruning efficiency and time complexity.

5 | Application to Human Activity Segmentation

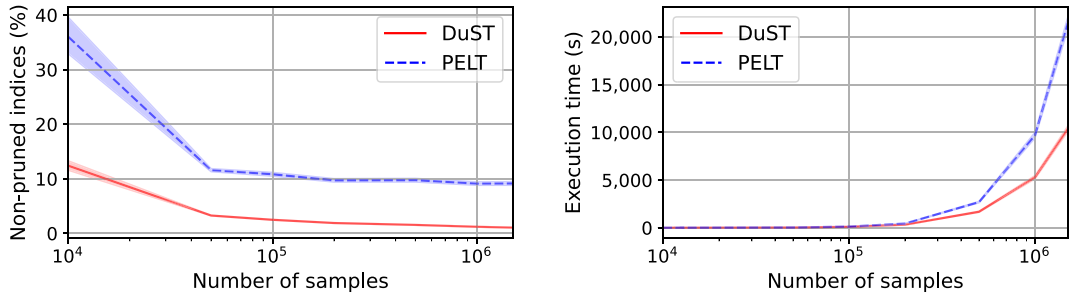
5.1 | Setting

We apply our method to Human Activity Sensing Consortium (HASC[2]) 2011 Challenge data (Kawaguchi et al. 2011). The challenge provided sensor data collected from seven individuals



(a) % of non-pruned indices (b) Execution time

FIGURE B2 | Pruning efficiency and execution time on the simulated data ('Few Changes, Few Symbols' setting).



(c) % of non-pruned indices (d) Execution time

FIGURE B3 | Pruning efficiency and execution time on the simulated data.

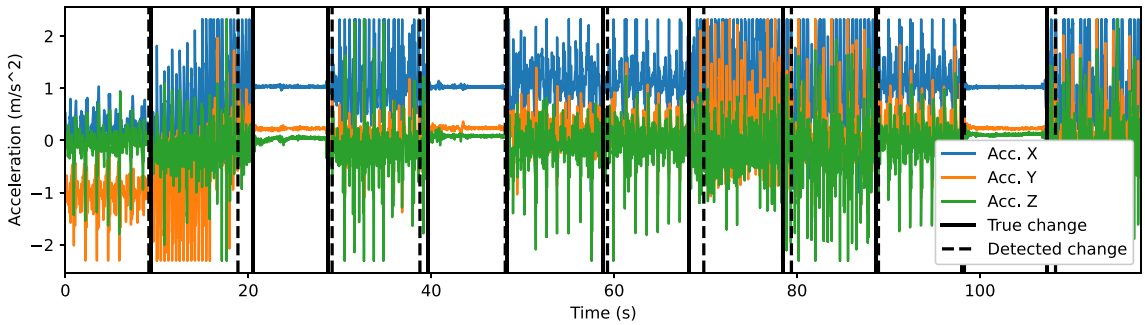


FIGURE B4 | Raw signal example for the human activity segmentation task (details in Section 5). The sequence of activities for this signal is walk, jog, stay, stairs down, stay, stairs up, walk, jog, skip, walk, stay and skip. Vertical solid lines indicate the true segmentation and the vertical dashed lines indicate the changes detected by DuST. Here, all changes are well detected.

engaging in various activities: walking, running, sitting, standing, stair ascent, stair descent and skipping. The sensors were wearable devices, for example, a smartphone, and recorded the participants' 3D acceleration and 3D angular velocity. There are 18 time series sampled at 100 Hz, lasting about 2 min each (12,000 samples). Figure B4 displays a signal example.

The signals are preprocessed as follows. Each acceleration component's short-term Fourier transform (STFT) is computed (window size of 3 s, with an overlap of 2.99 s). The STFTs are then concatenated to yield a 453-dimensional signal. A Gaussian mixture model (GMM) with 10 mixtures ($d = 10$) is fitted on the pooled samples. Finally, each sample is replaced by the probability of belonging to each Gaussian of the mixture, yielding a compositional signal. Similar pre-preprocessing procedures have been successfully applied in audio classification and speaker recognition (Malegaonkar, Ariyaeeinia, and Sivakumaran 2007; Hanifa, Isa, and Mohamad 2021).

The ARI and error on the number of predicted changes are reported, as well as the F1@1 s and F1@2 s as in Deldari et al. (2021). For both DuST and MS-CPD, the value of β is calibrated with a grid search on one signal, which is discarded for the rest of the study. The chosen value of β is the one in the grid that minimizes the error on the number of changes.

5.2 | Results

The segmentation results on the HASC data are given in Table B2:

TABLE B2 | Results on the human activity segmentation task.

Metric	DuST	MS-CPD
ARI	**0.88** (± **0.08**)	0.81 (± 0.08)
$\lvert\widehat{K} - K^\star\rvert$	**0.4** (± **1.0**)	1.2 (± 1.0)
F1@1 s	**88.0%** (± **13.4%**)	75.7% (± 17.2%)
F1@2 s	**94.8%** (± **8.6%**)	87.0% (± 11.2%)

Note: Here, $K^\star$ is the true number of change-points, and $\widehat{K}$ is the estimated one. For ARI and F1-scores, higher is better; for $\lvert\widehat{K} - K^\star\rvert$, lower is better. The best score is in bold.

- Our method outperforms MS-CPD according to all metrics. In particular, the number of changes is better estimated by DuST, which means that the penalty value calibrated on a separate signal generalizes better on new data. This is because Model 1 better approximate compositional time series, compared to the Gaussian assumption of MS-CPD.
- For a margin of 2 s, DuST detects the changes with high accuracy (F1@2 s is 94.8%), meaning that even after the symbolization procedure, it is possible to do change-point detection on the considerably summarized signal.
- The change the least precisely detected by DuST is the transition between 'walk' and 'stairs up' (3 misses out of 17 for a margin $M = 2$s). One example can be seen on the signal of Figure B4, around 60 s. This transition is arguably difficult to see on the raw data, and it is easy to imagine that the two activities have some similarities. Note that MS-CPD also struggles to detect this transition (11 misses out of 17).

6 | Conclusion

In this work, some new change-point estimators for compositional and categorical signals have been proposed. They are the solution to a discrete optimization problem, for which we give consistency guarantees and describe an efficient algorithm named `DuST`. This algorithm utilizes a dynamic programming approach and incorporates a pruning rule, allowing us to reduce the set of candidate change-point indices. Through simulation studies and real-world data set analysis, we have demonstrated that the `DuST` algorithm surpasses common methods in accuracy and speed.

Author Contributions

V.R. and C.T. designed methods, theoretical developments, and experiments. V.R. and C.T. wrote the main manuscript. C.T. conducted the simulation experiments and provided the software.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data that support the findings of this study are openly available in HASC Challenge 2011 at hasc.jp/hc2011.

Endnotes

1github.com/deepcharles/compositionaldust

2hasc.jp/hc2011/index-en.html

References

Alaee, S., R. Mercer, K. Kamgar, and E. Keogh. 2021. "Time Series Motifs Discovery Under DTW Allows More Robust Discovery of Conserved Structure." *Data Mining and Knowledge Discovery* 35: 863–910.

Arlot, S. 2019. "Minimal Penalties and the Slope Heuristics: A Survey." *Journal de la Société Française de Statistique* 160, no. 3: 1–160.

Barnett, I., and J.-P. Onnela. 2016. "Change Point Detection in Correlation Networks." *Scientific Reports* 6, no. 1: 18893.

Blotas, S., and C. Truong. 2024. "Structured Loss for Deep Change-Point Detection." In *Proceedings of the European Signal Processing Conference (Eusipco)*. Lyon, France.

Boyd, S., and L. Vandenberghe. 2004. *Convex Optimization*. Cambridge, Massachusetts, USA: Cambridge University Press.

Carl, G., G. Kesidis, R. R. Brooks, and S. Rai. 2006. "Denial-of-Service Attack-Detection Techniques." *IEEE Internet Computing* 10, no. 1: 82–89.

Chen, H., Y. Jia, G. Wang, and C. Zou. 2024. "Uncertainty Quantification for Data-Driven Change-Point Learning via Cross-Validation." In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 38, 11294–11301.

Cleynen, A., and E. Lebarbier. 2017. "Model Selection for the Segmentation of Multiparameter Exponential Family Distributions." *Electronic Journal of Statistics* 11, no. 1: 800–842.

Cohen, P., B. Heeringa, and N. Adams. 2002. "Unsupervised Segmentation of Categorical Time Series Into Episodes." In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, 99–106. Maebashi City, Japan.

Deldari, S., D. V. Smith, H. Xue, and F. D. Salim. 2021. "Time Series Change Point Detection With Self-Supervised Contrastive Predictive Coding." In *Proceedings of the World Wide Web Conference (WWW)*, 3124–3135. New York, NY, USA.

Fearnhead, P., and G. Rigaill. 2020. "Relating and Comparing Methods for Detecting Changes in Mean." *Stat* 9, no. 1: e291.

Fisher, T. J., J. Zhang, S. P. Colegate, and M. J. Vanni. 2022. "Detection and Modelling Changes in a Time Series of Proportions." *Annals of Applied Statistics* 16, no. 1: 477–494.

García, S., J. Luengo, J. A. Sáez, V. López, and F. Herrera. 2013. "A Survey of Discretization Techniques: Taxonomy and Empirical Analysis in Supervised Learning." *IEEE Transactions on Knowledge and Data Engineering* 25, no. 4: 734–750.

Germain, T., C. Truong, L. Oudre, and E. Krejci. 2023. "Unsupervised Classification of Plethysmography Signals With Advanced Visual Representations." *Frontiers in Physiology* 14: 1154328.

Godichon-Baggioni, A., C. Maugis-Rabusseau, and A. Rau. 2019. "Clustering Transformed Compositional Data Using K-Means, With Applications in Gene Expression and Bicycle Sharing System Data." *Journal of Applied Statistics* 46, no. 1: 47–65.

Hanifa, R. M., K. Isa, and S. Mohamad. 2021. "A Review on Speaker Recognition: Technology and Challenges." *Computers & Electrical Engineering* 90: 107005.

Hocking, T. D., G. Rigaill, P. Fearnhead, and G. Bourque. 2020. "Constrained Dynamic Programming and Supervised Penalty Learning Algorithms for Peak Detection in Genomic Data." *Journal of Machine Learning Research (JMLR)* 21: 1–40.

Hubert, L., and P. Arabie. 1985. "Comparing Partitions." *Journal of Classification* 2: 193–218.

Jackson, B., J. D. Scargle, D. Barnes, S. Arabhi, A. Alt, P. Gioumousis, E. Gwin, P. Sangtrakulcharoen, L. Tan, and T. T. Tsai. 2005. "An Algorithm for Optimal Partitioning of Data on an Interval." *IEEE Signal Processing Letters* 12, no. 2: 105–108.

James, N. A., and D. S. Matteson. 2013. "ecp: An R Package for Nonparametric Multiple Change Point Analysis of Multivariate Data." arXiv preprint arXiv:1309.3295.

Jian, S., G. Pang, L. Cao, K. Lu, and H. Gao. 2019. "CURE: Flexible Categorical Data Representation by Hierarchical Coupling Learning." *IEEE Transactions on Knowledge and Data Engineering* 31, no. 5: 853–866.

Jung, S., L. Oudre, C. Truong, E. Dorveaux, L. Gorintin, N. Vayatis, and D. Ricard. 2021. "Adaptive Change-Point Detection for Studying Human Locomotion." In *Proceedings of the International Conference of the IEEE Engineering in Medecine and Biology Society (EMBC)*. Guadalajara, Mexico.

Kawaguchi, N., N. Ogawa, Y. Iwasaki, K. Kaji, T. Terada, K. Murao, S. Inoue, Y. Kawahara, Y. Sumi, and N. Nishio. 2011. "HASC Challenge: Gathering Large Scale Human Activity Corpus for the Real-World Activity Understandings." In *Proceedings of the Augmented Human International Conference*, 1–5. Tokyo, Japan.

Kelil, A., and S. Wang. 2008. "SCS: A New Similarity Measure for Categorical Sequences." In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, 343–352. Pisa, Italy.

Killick, R., P. Fearnhead, and I. A. Eckley. 2012. "Optimal Detection of Changepoints With a Linear Computational Cost." *Journal of the American Statistical Association* 107, no. 500: 1590–1598.

Lavielle, M. 1999. "Detection of Multiples Changes in a Sequence of Dependant Variables." *Stochastic Processes and Their Applications* 83, no. 1: 79–102.

Lebarbier, E. 2005. "Detecting Multiple Change-Points in the Mean of Gaussian Process by Model Selection." *Signal Processing* 85: 717–736.

Li, X., and J. Lin. 2017. "Linear Time Complexity Time Series Classification With Bag-of-Pattern-Features." In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, 277–286. New Orleans, LA, USA.

Londschien, M., S. Kovács, and P. Bühlmann. 2021. "Change-Point Detection for Graphical Models in the Presence of Missing Values." *Journal of Computational and Graphical Statistics* 30, no. 3: 768–779.

Lung-Yut-Fong, A., C. Lévy-Leduc, and O. Cappé. 2011. "Robust Changepoint Detection Based on Multivariate Rank Statistics." In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3608–3611. Prague, Czech Republic.

Maidstone, R., T. Hocking, G. Rigaill, and P. Fearnhead. 2017. "On Optimal Multiple Changepoint Algorithms for Large Data." *Statistics and Computing* 27: 519–533.

Malegaonkar, A. S., A. M. Ariyaeeinia, and P. Sivakumaran. 2007. "Efficient Speaker Change Detection Using Adapted Gaussian Mixture Models." *IEEE Transactions on Audio, Speech and Language Processing* 15, no. 6: 1859–1869.

Pawlowsky-Glahn, V., and J. J. Egozcue. 2016. "Spatial Analysis of Compositional Data: A Historical Review." *Journal of Geochemical Exploration* 164: 28–32.

Pein, F., and R. D. Shah. 2021. "Cross-Validation for Change-Point Regression: Pitfalls and Solutions." arXiv e-prints arXiv:2112.03220.

Pishchagina, L., G. Rigaill, and V. Runge. 2023. "Geometric-Based Pruning Rules for Change Point Detection in Multiple Independent Time Series."

Prabuchandran, K. J., N. Singh, P. Dayama, A. Agarwal, and V. Pandit. 2022. "Change Point Detection for Compositional Multivariate Data." *Applied Intelligence* 52, no. 2: 1930–1955.

Rieser, C., and P. Filzmoser. 2023. "Extending Compositional Data Analysis From a Graph Signal Processing Perspective." *Journal of Multivariate Analysis* 198: 105209.

Runge, V. 2020. "Is a Finite Intersection of Balls Covered by a Finite Union of Balls in Euclidean Spaces?."*Journal of Optimization Theory and Applications* 187, no. 2: 431–447.

Seichepine, N., S. Essid, C. Fevotte, and O. Cappé. 2014. "Piecewise Constant Nonnegative Matrix Factorization." In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6721–6725. Florence, Italy.

Sundqvist, M., J. Chiquet, and G. Rigaill. 2022. "Adjusting the Adjusted Rand Index: A Multinomial Story." *Computational Statistics* 38, no. 1: 327–347.

Tartakovsky, A., I. Nikiforov, and M. Basseville. 2014. *Sequential Analysis: Hypothesis Testing and Changepoint Detection*. Boca Raton, FL: CRC Press.

Truong, C., L. Oudre, and N. Vayatis. 2020. "Selective Review of Offline Change Point Detection Methods." *Signal Processing* 167: 107299.

Ueda, K., Y. Ike, and K. Yamanishi. 2022. "Change Detection With Probabilistic Models on Persistence Diagrams." In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, 1191–1196. Orlando, FL, United States.

Verzelen, N., M. Fromont, M. Lerasle, and P. Reynaud-Bouret. 2020. "Optimal Change-Point Detection and Localization." *The Annals of Statistics* 51: 1586–1610. https://arxiv.org/abs/2010.11470.

Wang, G., C. Zou, and G. Yin. 2018. "Change-Point Detection in Multinomial Data With a Large Number of Categories." *The Annals of Statistics* 46, no. 5: 2020–2044.

Warrens, M. J., and H. van der Hoef. 2022. "Understanding the Adjusted Rand Index and Other Partition Comparison Indices Based on Counting Object Pairs." *Journal of Classification* 39, no. 3: 487–509.

Wu, S., and S. Wang. 2013. "Information-Theoretic Outlier Detection for Large-Scale Categorical Data." *IEEE Transactions on Knowledge and Data Engineering* 25, no. 3: 589–602.

Yao, Y.-C. 1988. "Estimating the Number of Change-Points via Schwarz' Criterion." *Statistics & Probability Letters* 6, no. 3: 181–189.

Zhang, N., and S. David. 2007. "A Modified Bayes Information Criterion With Applications to the Analysis of Comparative Genomic Hybridization Data." *Biometrics* 63: 22–32.

Zou, C., G. Wang, and R. Li. 2020. "Consistent Selection of the Number of Change-Points via Sample-Splitting." *The Annals of Statistics* 48, no. 1: 413–439.

Appendix A

Proof of Theorem 1

Theorem 1 is an application of Theorem 3.1 from Lavielle (1999), which is stated in a more general context. We only need to show that the assumptions on the cost function $c(\cdot, \cdot)$ (also called a contrast function) and the 'noise' process $\boldsymbol{z}_{0..n}(\boldsymbol{p})$ are verified. For brevity, we use the notations of Lavielle (1999). Note that there is a scale factor of $1/n$ between the β_n in Lavielle (1999) and ours.

A.1 | Assumptions on the Contrast Function (Assumption H1 [Lavielle, (1999)])

The contrast function can be decomposed as follows:

$$c(\boldsymbol{y}_{a..b}, \boldsymbol{p}) = \sum_{t=a+1}^{b} \psi(\boldsymbol{p})^\top \xi(\boldsymbol{y}_t) \text{ where } \psi(\boldsymbol{p}) := -\log \boldsymbol{p} \text{ and } \xi(\boldsymbol{y}) := \boldsymbol{y}. \tag{A1}$$

Define $w(\boldsymbol{p}, \boldsymbol{q}) := -\boldsymbol{p}^\top \log \boldsymbol{q}$. Then $v(\boldsymbol{p}, \boldsymbol{q}) := w(\boldsymbol{p}, \boldsymbol{q}) - w(\boldsymbol{p}, \boldsymbol{p}) = \boldsymbol{p}^\top \log(\boldsymbol{p}/\boldsymbol{q})$ (where the division is elementwise) is the Kullback-Leibler (KL) divergence between the discrete probability distributions $\boldsymbol{p}$ and $\boldsymbol{q}$. The KL divergence satisfies the remaining requirements of Assumption H1 (Lavielle 1999), namely, (i) $v(\boldsymbol{p}, \boldsymbol{q}) \geq 0$, (ii) it is equal to 0 if and only if $\boldsymbol{p} = \boldsymbol{q}$ and (iii) $v(\boldsymbol{p}, \boldsymbol{q}) \geq 2\|\boldsymbol{p} - \boldsymbol{q}\|_2^2$.

A.2 | Assumptions on the Fluctuations of the Contrast Function (Assumption H2 [Lavielle, (1999)])

Assumption 1 is equivalent to Assumption H2 of Lavielle (1999) where, in the notations of Lavielle (1999), $\boldsymbol{z}_{0..n}(\boldsymbol{p})$ is $\eta(\boldsymbol{p})$.

Appendix B

Proof of Proposition 2

Note that if $s' < s < t$, then $\mathbf{S}_{s'..t} - \mathbf{S}_{s..t} = \mathbf{S}_{s'..s}$. We write down the Lagrangian of optimization problem (16):

$$\mathcal{L}(\boldsymbol{\theta}, \mu, \lambda) := V_s - \sum_{u=s+1}^{t} \boldsymbol{\theta}^\top \boldsymbol{y}_u + \beta + \mu\left(V_s - \sum_{u=s+1}^{t} \boldsymbol{\theta}^\top \boldsymbol{y}_u - V_{s'} + \sum_{u=s'+1}^{t} \boldsymbol{\theta}^\top \boldsymbol{y}_u\right) + \lambda\left(\sum_{i=1}^{d} e^{\theta_i} - 1\right), \tag{B1}$$

which leads to

$$\mathcal{L}(\boldsymbol{\theta}, \mu, \lambda) := V_s - \mu(V_{s'} - V_s) + \beta - \boldsymbol{\theta}^\top\left(\mathbf{S}_{s..t} - \mu \mathbf{S}_{s'..s}\right) + \lambda\left(\sum_{i=1}^{d} e^{\theta_i} - 1\right), \tag{B2}$$

for any $\mu \geq 0$ and $\lambda \in \mathbb{R}$. The critical point $\boldsymbol{\theta}^\star$ of $\mathcal{L}(\boldsymbol{\theta}, \mu, \lambda)$ is given by relation $e^{\theta_i^\star} = (1/\lambda)\left[S_{s..t,i} - \mu S_{s'..s,i}\right]$. To have $\theta_i^\star$ inside the set Θ, we need additional conditions on the multipliers. First, the Lagrange multiplier λ is chosen such that the quantities

$\exp(\theta_i^\star)$ sum to 1. Denote by $\lambda^\star$ this value: $\lambda^\star := (t-s) - \mu(s-s')$. Second, to keep the quantities $e^{\theta_i^\star}$ positive, μ must satisfy for all $i = 1, \ldots, d$, $S_{s..t,i} - \mu S_{s'..s,i} \geq 0$. This latter condition defines our bound $\overline{\mu}$ for parameter μ (18). Expression for $\mathcal{L}(\boldsymbol{\theta}^\star, \mu, \lambda^\star)$ is equal to the Lagrangian dual function $\mathcal{L}(\mu, \lambda^\star)$ (17) and the inequality test $\mathcal{L}(\mu, \lambda^\star) > V_t + \beta$ leads to expression (18) as expected.