A review of change point detection methods[☆]

C. Truong[a,b,*], L. Oudre[c], N. Vayatis[a,b]

[a] *CMLA, ENS Cachan, CNRS, Université Paris-Saclay, 94235, Cachan, France*
[b] *Cognac G, University Paris Descartes, CNRS, 75006 Paris, France*
[c] *L2TI, University Paris 13, 93430 Villetaneuse, France*

Abstract

In this work, methods to detect one or several change points in multivariate time series are reviewed. They include retrospective (off-line) procedure such as maximum likelihood estimation, regression, kernel methods, etc. In this large area of research, applications are numerous and diverse; many different models and operational constraints (on precision, complexity,...) exist. A formal framework for change point detection is introduced to give sens to this significant body of work. Precisely, all methods are described as a collection of three elements: a cost function, a search method and a constraint on the number of changes to detect. For a given method, we detail the assumed signal model, the associated algorithm, theoretical guarantees (if any) and the application domain. This approach is intended to facilitate prototyping of change point detection methods: for a given segmentation task, one can appropriately choose among the described elements to design an algorithm.

Keywords: change point detection, signal segmentation, review, non-parametric method, signal processing

1. Introduction

Segmenting time series is central to a wide range of applications. Many real-world data streams are made of consecutive regimes separated by an abrupt change. In such situations, the underlying model producing the data switches multiple times. Some regimes can be of no use or even interfere with signal processing algorithms. This particular issue arises in contexts ranging from speech recognition to medical monitoring [22, 28, 33, 72, 107].

2. General setting

Before reviewing methods from the literature, the change point detection problem is formally introduced, especially the general setting and the methodology developed in this work. To ease notations, we make the following shorthands:

[☆] This work was supported by a public grant as part of the Investissement d'avenir project, reference ANR-11-LABX-0056-LMH, LabEx LMH.

[*] Corresponding author

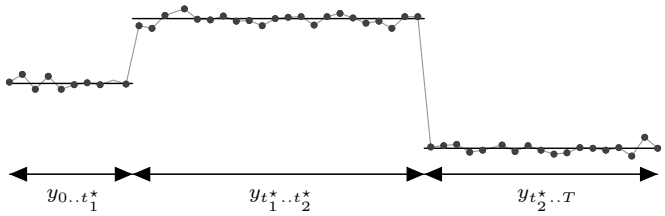

Figure 1: Signal example $y = \{y_t\}_{t=1}^{100}$ with two change points $t_1^\star$ and $t_2^\star$.

- for a given signal $y = \{y_t\}_{t=1}^{T}$, the $(b-a)$-sample long sub-signal $\{y_t\}_{t=a+1}^{b}$ $(1 \leq a < b \leq T)$ is simply denoted $y_{a..b}$; the complete signal is therefore $y = y_{0..T}$;
- a set of indexes is denoted by a bold letter: $\mathbf{t} = \{t_1, t_2, \dots\} \subset \{1, \dots, T\}$, and its cardinal is $|\mathbf{t}|$;
- for a set of indexes $\mathbf{t} = \{t_1, \dots, t_K\}$, the dummy indexes $t_0 := 0$ and $t_{K+1} := T$ are implicitly available;
- for a given index (of a change point) t $(1 \leq t \leq T)$, the associated *fraction* (of a change point) is $\tau := t/T \in (0, 1]$;
- a set of fractions is denoted by a bold greek letter: $\boldsymbol{\tau} = \{\tau_1, \tau_2, \dots\}$, and its cardinal is $|\boldsymbol{\tau}|$.

Notations are illustrated on Figure 1.

2.1. Problem statement

In the the offline or retrospective change point detection framework, we consider a non-stationary random process $y = \{y_1, \dots, y_T\}$ that takes value in $\mathbb{R}^d$ $(d \geq 1)$. The signal y is assumed to be piecewise stationary, meaning that some characteristics of the process change abruptly at some unknown instants $t_1^\star < t_2^\star < \cdots < t_K^\star$. The corresponding segmentation is $\mathbf{t}^\star = \{t_1^\star, \dots, t_{K^\star}^\star\}$. In other words, the sub-signals $y_{t_k^\star..t_{k+1}^\star}$ are homogeneous.
Further assume that there exists a set $\boldsymbol{\tau}^\star = \{\tau_1^\star, \dots, \tau_{K^\star}^\star\}$ of $K^\star$ change point fractions such that

$$t_k^\star = \lfloor T\tau_k^\star \rfloor. \tag{1}$$

Change point detection consists in estimating those instants $t_k^\star$ when a particular realization of y is observed. A change point detection algorithm returns an estimated segmentation, usually denoted $\hat{\mathbf{t}} = \{\hat{t}_1, \hat{t}_2, \dots\}$ with $1 \leq \hat{t}_1 < \hat{t}_2 < \cdots \leq T$. Note that the number of changes $K^\star$ is not necessarily known and must sometimes be estimated.
Quality of a change point detection method is measured by the distance between an estimation and the true change point index, normalized by the number T of samples:

$$\forall k, \quad |\hat{t}_k - t_k^\star|/T \tag{2}$$

Ideally, an algorithm should have this error decreasing to zero when the number T of samples is growing to infinity. This property is called *asymptotic consistency*. This notion is formally introduced in the context of change point detection.

Definition 1 (Asymptotic consistency)**.** *A change point detection algorithm is said to be asymptotically consistent if the estimated segmentation* $\hat{\mathbf{t}}$ *is such that* $|\hat{\mathbf{t}}| = K^\star$ *and*

$$\max_{k=1,\dots,K^\star} |\hat{t}_k - t_k^\star|/T \xrightarrow{p} 0. \tag{3}$$

where the $\hat{t}_k$ *are the elements of* $\hat{\mathbf{t}}$ *in ascending order.*

Remark 1. *In Definition 1, the change point fractions are consistent, and not the indexes themselves. Distances* $|\hat{t}_k - t_k^\star|$ *between true change point indexes and their estimated counterparts do not converge to 0, even for simple models [6, 7, 17, 26, 30, 129]. As a result, consistency results in the literature only deal with change point fractions.*

2.2. Methodology

In this work, we consider a general class of algorithms which encompasses a large part of the literature [2, 4–7, 9, 18, 22, 24, 25, 27, 33, 44, 49, 54, 58, 65–67, 70, 72, 72, 78, 81, 90, 99, 100, 117, 118]. Those methods adhere to or are an approximation of a general format where a suitable contrast function $V(\cdot)$ is minimized. The contrast function is expressed as a sum of segment costs:

$$V(\mathbf{t}, y) := \sum_{k=0}^{K} c(y_{t_k..t_{k+1}}) \tag{4}$$

where $c(\cdot)$ is a cost function which measures goodness-of-fit of the sub-signal $y_{t_k..t_{k+1}} = \{y_t\}_{t_k+1}^{t_{k+1}}$ to a specific model. The cost function can for instance be the sum of squared residuals after a linear fit [18]: its value is expected to be low on "homogeneous" sub-signals (here, linear sub-signals) and large on sub-signals that are not well-approximated by the linear model (e.g. if they contain one or more breaks).
The sum of costs $V(\mathbf{t}, y)$ is denoted $V(\mathbf{t})$ when it is obvious from the context that it refers to the signal y.
Change point detection methods aim at minimizing $V(\cdot)$ under certain constraints. They mostly fall into two categories, depending on whether the number of change points is fixed:

- if K is fixed, the change point estimation $\hat{\mathbf{t}}$ is the minimizer of the discrete optimization problem

$$\min_{|\mathbf{t}|=K} V(\mathbf{t}); \tag{5}$$

- if K is not fixed, the change point estimation $\hat{\mathbf{t}}$ is the minimizer of the discrete optimization problem

$$\min_{\mathbf{t}} V(\mathbf{t}) + \text{pen}(\mathbf{t}) \tag{6}$$

  where $\text{pen}(\mathbf{t})$ is an appropriate measure of the complexity of a segmentation $\mathbf{t}$.

Under this methodology, a change point detection procedure consists in setting the three following elements:

- choosing the relevant cost function $c(\cdot)$ (4) to measure homogeneity in the sub-signals, which is related to the type of change to detect;

- solving the discrete optimization problems (5) and (6), exactly or appropriately;
- constraining the number of change points, by fixing it to K (5) or penalizing it with $\mathrm{pen}(\cdot)$ (6).

The remainder of this chapter is organized as follows. Cost functions from the literature are first presented in Section 3. Then Section 4 describes search methods that solve the change point detection problem with a known number of segments. After, search methods when K is unknown are presented in Section 5. Section 6 lists other types of methods from the literature. All methods are summarized in Section 7 and the main algorithms are detailed in pseudo-code in Section 9.

3. Cost functions

We first present the main cost functions used in the literature. They are related to the assumed underlying model. In the following, the cost functions' are applied on the general sub-signal $y_{a..b}$ for $1 \leq a < b \leq T$. (Certain cost functions might require the sub-signal $y_{a..b}$ to contain strictly more than one sample, i.e. $b - a \geq 2$.)

3.1. Piecewise i.i.d. signals

We first describe the general parametric model where the signal y is modelled by a i.i.d. variables with piecewise constant distribution. This formulation encompasses a large number of change point models [1, 14, 32–34, 72, 81, 99, 100, 115, 125].
Assume that $y_1, \dots, y_T$ are independent random variables, with probability distribution $f(y_t|\theta)$ depending on the vector-valued parameter θ. This parameter θ represents a quantity of interest whose value changes abruptly at unknown instants that are to be estimated. Formally, assume that there exists a set of true change points $\mathbf{t}^\star = \{t_1^\star, \dots\}$ and that

$$y_t \sim \sum_{i=0}^{K} f(\cdot|\theta_k) \mathbb{1}(t_k^\star < t \leq t_{k+1}^\star). \tag{7}$$

In this context, change point detection is performed through maximum likelihood estimation. The related cost function is the negative log-likelihood and is given by

$$c_{\text{i.i.d.}}(y_{a..b}) := -\sup_{\theta} \sum_{t=a+1}^{b} \log f(y_t|\theta). \tag{8}$$

Choice of a distribution is motivated by prior knowledge on the data. Historically, the Gaussian distribution was often used to model mean and scale shifts [5, 82, 85, 91, 123]. A large part of the literature then evolved towards other parametric distributions, most notably resorting to distributions from the general exponential family [48, 50, 99].

Example 1 (Mean-shifts for normal data)**.** *The mean-shift model is one of the most studied model in the change point detection literature [35, 96, 101, 108, 123]. Let $y_1, \dots, y_T$ be a sequence of independent normal random variables with piecewise constant mean and constant variance. In this context, the cost function $c_{\text{i.i.d.}}(\cdot)$ (8) becomes*

$$c_{L_2}(y_{a..b}) := \sum_{t=a+1}^{b} \|y_t - \bar{y}_{a..b}\|_2^2 \tag{9}$$

where $\bar{y}_{a..b}$ is the empirical mean of the sub-signal $y_{a..b}$. This cost function is also referred to as the quadratic error loss. It has been applied for instance on DNA array data [50], geology signals [33].

Example 2 (Mean-shifts and scale-shifts for normal data)**.** *A natural extension to the mean-shift model is to allow the variance to abruptly change as well. Let $y_1, \dots, y_T$ be a sequence of independent normal random variables with piecewise constant mean and variance. In this context, the cost function $c_{\text{i.i.d.}}(\cdot)$ (8) becomes*

$$c_{\Sigma}(y_{a..b}) := \log\det \widehat{\Sigma}_{a..b} \tag{10}$$

where $\widehat{\Sigma}_{a..b}$ is the empirical covariance matrix of the sub-signal $y_{a..b}$. This cost function can be used to detect changes in the first two moments of random (not necessarily Gaussian) variables, even though it is the Gaussian likelihood that is plugged in $c_{\text{i.i.d.}}(\cdot)$ [72, 89]. It has been applied for instance on stock market time series [89] and biomedical data [33].

Example 3 (Count data)**.** *Let $y_1, \dots, y_T$ be a sequence of independent Poisson distributed random variables with piecewise constant rate parameter. The cost function $c_{\text{Poisson}}(\cdot)$ associated with this model is*

$$c_{\text{Poisson}}(y_{a..b}) := -(b-a)\bar{y}_{a..b}\log\bar{y}_{a..b} \tag{11}$$

where $\bar{y}_{a..b}$ is the empirical mean of the sub-signal $y_{a..b}$. This model has been applied for instance to model count data [36, 82].

Theoretically, change point detection verifies consistency properties as a maximum likelihood estimation method. Indeed, it has been shown that, "under extremely general conditions" [89], the estimated optimal change point fractions converge in probability to the true ones as the number grows to infinity. Change point detection with the cost function $c_{\text{i.i.d.}}$ is asymptotically consistent (Definition 1). Note that the change point indexes $\hat{t}_k$ are not consistently estimated: only the change point fractions $\hat{t}_k/T$ are. In fact, the estimated indexes do not converge in probability to the true change point indexes [18, 90].

Remark 2. *A slightly more general model can be formulated by letting the distribution function $f(\cdot|\theta)$ change over over time. This can in particular model the presence of unwanted changes in the statistical properties of the signal (for instance in the statistical structure of the noise [89]). The function $f(\cdot|\theta)$ is replaced in (7) by a sequence of distribution functions $f_t(\cdot|\theta)$ which are not assumed to be identical for all indexes t. Changes in the functions f_t are considered nuisance parameters and we are only interested in the variations of the parameter θ. Nevertheless, similar properties (about estimation consistency) are obtained in the context of (7) or its generalization, "under extremely general conditions" [90]. We refer the reader to [89, 90] for theoretical results in this context.*

To sum up, parametric cost functions of the form (8) are useful because there are theoretical consistency guarantees [18, 75, 90]. In addition, they can accommodate a wide range of situations. However, if the data are not well approximated by any of the available parametric distributions, change point detection performance is greatly impacted, motivating the use of other models [1, 55, 63, 79, 113].

3.2. Linear model

Linear models have been introduced and extensively studied in the context of change point detection by a series of important contributions [6–8, 10–16, 16, 111]. Both the theoretical and

algorithmic aspects of linear models are tackled in those papers.
Linear models are used when there exists a linear relationship between variables that changes abruptly at some unknown instants. There are often referred to as "structural changes" in the related literature [6–8]. This formulation generalizes several well-known models such as the autoregressive (AR) model [2, 13], multiple regressions [8, 10], etc.
Mathematically, the signal y is regarded as a (univariate) response variable; two signals of covariates $x = \{x_t\}_{t=1}^T$ and $z = \{z_t\}_{t=1}^T$ are introduced, respectively $\mathbb{R}^p$-valued and $\mathbb{R}^q$-valued. Assume that a true underlying set $\mathbf{t}^\star = \{t_1^\star, \dots, t_K^\star\}$ of change points exists. The general linear model [111, 115] is given by:

$$\forall\, t,\ t_k^\star < t \leq t_{k+1}^\star, \quad y_t = x_t' u_k + z_t' v + \varepsilon_t \quad (k = 0, \dots, K) \tag{12}$$

where the $v \in \mathbb{R}^q$ and $u_k \in \mathbb{R}^p$ are unknown regression parameters and ε_t is noise. In this model (12), also known as a partial structural change model: the linear relationship between y and x changes abruptly, while the linear relationship between y and z is constant. The pure structural change model is obtained by removing the term $z_t' v$ from Equation 12.
The related cost function is based on the least-squares residuals and is given by:

$$c_{\text{linear}}(y_{a..b}) := \min_{u \in \mathbb{R}^p, v \in \mathbb{R}^q} \sum_{t=a+1}^{b} (y_t - x_t' u - z_t' v)^2. \tag{13}$$

A closed-form formula exists as it is a simple least-square regression.
Theoretically, the estimated change points asymptotically converge in probability to the true change points (Definition 1) under mild assumptions on the distribution on the covariates (x_t, z_t), distribution of the noise ε_t and the minimal distance between two change points $t_{k+1}^\star - t_k^\star$ (see [111] for details). Additional asymptotic properties are also provided [16, 111, 115], for instance the asymptotic distribution of the change point locations and the limiting distribution of $V(\hat{t}_K)$ (which can be used to design a statistical test).

Remark 3 (Least absolute deviation)**.** *The cost function* (13) *is sometimes replaced by the sum of the absolute deviations [7, 11]:*

$$c_{\text{linear},L_1}(y_{a..b}) := \min_{u \in \mathbb{R}^p, v \in \mathbb{R}^q} \sum_{t=a+1}^{b} |y_t - x_t' u - z_t' v|. \tag{14}$$

Theoretically, asymptotic consistency of change point estimates still holds. Using this cost also provides better performance for certain noise distributions [72, 99]. (Nevertheless, the least square function is more efficient for a zero-mean Gaussian i.i.d. noise.). However, practically, it is computationally intensive to solve this minimization problem.

Practically, this model is often applied on economic and financial data [6–8]. More recently, this model has for instance been used to show that the predictive linear model of stock returns can change abruptly [110]. In addition, breaks in the "comovement" of several economic parameters of the G-7 growth have been investigated using the linear model [43]. In [14], it has been applied on panel data.

Example 4 (Autoregressive model)**.** *Piecewise autoregressive models falls into the general formulation of linear models. Indeed, by setting* $x_t = [y_{t-1}, y_{t-2}, \dots, y_{t-p}]$ *in the linear model* (12) *and*

removing the term $z_t' v$ (yielding a pure structural change model), the signal y is in effect piecewise autoregressive with order p. The resulting cost function, denoted $c_{AR}(\cdot)$ is able to detect shifts in the autoregressive coefficients of a process [13, 31]. Piecewise autoregressive processes are a special case of time-varying ARMA processes.
Theoretical consistency guarantees hold even if the noise ε_t is correlated, for instance a moving-average process [13, 115]. Therefore, in practice, change point detection can be applied on full ARMA processes.
The piecewise autoregressive model has been applied on piecewise weakly stationarity signals, such as EEG/ECG time series [115], functional magnetic resonance imaging (fMRI) time series [104] and speech recognition tasks [2].

3.3. Kernel change point detection

Change point detection can be performed on a high-dimensional mapping of the original signal, implicitly defined by a kernel function [4, 42, 59, 62]. This technique is central to many machine learning developments such as support vector machine or clustering [56, 120].
The original signal y is mapped onto a reproducing Hilbert space (RKHS) $\mathcal{H}$ associated with a kernel function $k(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$. The related mapping function $\phi : \mathbb{R} \to \mathcal{H}$ is implicitly defined by

$$\phi(y_t) = k(y_t, \cdot) \in \mathcal{H} \quad \text{and} \quad \langle \phi(y_s) | \phi(y_t) \rangle_{\mathcal{H}} = k(y_s, y_t) \tag{15}$$

for any samples $y_s, y_t \in \mathbb{R}^d$. The RKHS norm $\|\cdot\|_{\mathcal{H}}$ is also implicitly defined by

$$\|\phi(y_t)\|_{\mathcal{H}}^2 = k(y_t, y_t). \tag{16}$$

Intuitively, the signal is expected to become piecewise constant once mapped to this high-dimensional (possibly infinite dimensional) feature space $\mathcal{H}$. This is for instance true for a signal of independent random variables with piecewise constant distribution (described in (7)), under some assumptions on the kernel $k(\cdot,\cdot)$ (see Remark 4). The objective is then to detect mean-shifts in the embedded signal. The related cost function measures the "average scatter" [59] and is given by

$$c_{\text{kernel}}(y_{a..b}) := \sum_{t=a+1}^{b} \|\phi(y_t) - \bar{\mu}_{a..b}\|_{\mathcal{H}}^2 \tag{17}$$

where $\bar{\mu}_{a..b} \in \mathcal{H}$ is the empirical mean of the embedded signal $\{\phi(y_t)\}_{t=a+1}^{b}$. Thanks to the well-known "kernel trick", the explicit computation of the mapped data samples is not required. Indeed, after simple algebraic manipulations, the kernel cost function (17) can be rewritten:

$$c_{\text{kernel}}(y_{a..b}) = \sum_{t=a+1}^{b} k(y_t, y_t) \; - \; \frac{1}{b-a} \sum_{s,t=a+1}^{b} k(y_s, y_t). \tag{18}$$

Any kernel can be plugged into this expression [56, 57], but the most common kernel is the Gaussian kernel (or radial basis function):

$$c_{\text{rbf}}(y_{a..b}) := (b-a) \; - \; \frac{1}{b-a} \sum_{s,t=a+1}^{b} \exp(-\gamma \|y_s - y_t\|^2) \tag{19}$$

where $\gamma > 0$ is the so-called bandwidth parameter. Using the linear kernel in this context is formally equivalent to using the quadratic error loss $c_{L_2}(\cdot)$ described in Example 1 for piecewise constant signals.

Remark 4 (Probability measures and RKHS embeddings)**.** *Under certain conditions on the kernel $k(\cdot,\cdot)$, the mapping $\phi(\cdot)$ transforms piecewise i.i.d. signals into piecewise constant signals (in the feature space $\mathcal{H}$). Here, we assume that the kernel $k(\cdot,\cdot)$ is translation invariant, meaning that $k(y_s, y_t) = \psi(y_s - y_t)\ \forall s,t$, where ψ is a bounded continuous positive definite function on $\mathbb{R}^d$ [126, Assumption 1].*
Let $\mathbb{P}$ denote a probability distribution defined over $\mathbb{R}^d$. Then there exists a unique element $\mu_{\mathbb{P}} \in \mathcal{H}$ [126, Theorem 7], called the mean embedding (of $\mathbb{P}$), such that

$$\mu_{\mathbb{P}} = \mathbb{E}_{X\sim\mathbb{P}}\left[\phi(X)\right]. \tag{20}$$

In addition, the mapping $\mathbb{P} \mapsto \mu_{\mathbb{P}}$ is injective. In other words,

$$\mu_{\mathbb{P}} = \mu_{\mathbb{Q}} \iff \mathbb{P} = \mathbb{Q}. \tag{21}$$

where $\mathbb{Q}$ denotes a probability distribution defined over $\mathbb{R}^d$.
Intuitively, it follows from Equation 20 that any $\mathbb{R}^d$-valued random variable $X \sim \mathbb{P}$ is mapped by $\phi(\cdot)$ to a $\mathcal{H}$-valued random variable with mean equal to $\mu_{\mathbb{P}}$, the mean embedding of $\mathbb{P}$. Moreover, because this mean embedding is injective, two random variables with different distributions are mapped to two random variables with different means. To put it another way, a piecewise i.i.d. signal is mapped by $\phi(\cdot)$ to a random signal with piecewise constant mean. As a consequence, distribution changes in the original space are transformed into mean-shifts in the feature space $\mathcal{H}$.

Theoretically, the asymptotic consistency of the change point estimates has not been proven in the literature (except if the kernel is linear). A result on the convergence of the sum of costs $V(\hat{\mathbf{t}}_K)$ to $V(\mathbf{t}^\star)$ can be found in [4].
To sum-up, kernel change point detection is a non-parametric and model-free method. It can be used in the general setting where the signal to segment is piecewise i.i.d. without prior knowledge on the form of the underlying distribution. This method was shown to be competitive in many different settings without much tuning. For instance, the cost function (17) was applied on physiological data streams such as the Brain-Computer Interface (BCI) [59] and on a video time series segmentation task [4].

3.4. Mahalanobis-type metric

The quadratic error loss (9), adapted for detecting mean-shifts (Examples 1), can be replaced by a Mahalanobis-type pseudo-metric. In the context of change point detection, it has been used along with a metric learning algorithm [87] to adapt the metric to the considered signal. This is also a common occurrence in clustering methods [40, 64, 133].
Formally, for any symmetric positive semi-definite matrix $M \in \mathbb{R}^{d\times d}$, the associated pseudo-metric $\|\cdot\|_M$ is given by:

$$\|y_t - y_s\|_M^2 := (y_t - y_s)' M (y_t - y_s) \tag{22}$$

for any two samples $y_s,\ y_t$. The resulting cost function is defined as follows:

$$c_M(y_{a..b}) := \sum_{t=a+1}^{b} \|y_t - \bar{y}_{a..b}\|_M^2 \tag{23}$$

where $\bar{y}_{a..b}$ is the empirical mean of the sub-signal $y_{a..b}$.
Original, the metric matrix M was set equal to the inverse of the covariance matrix, yielding the Mahalanobis metric [98]:

$$M = \widehat{\Sigma}^{-1} \tag{24}$$

where $\widehat{\Sigma}$ is the empirical covariance matrix of the signal y.
Intuitively, measuring distances with the pseudo-norm $\|\cdot\|_M$ is equivalent to applying a linear transformation on the data and using the regular (Euclidean) norm $\|\cdot\|$. Indeed, decomposing the matrix $M = U'U$ yields:

$$\|y_t - y_s\|_M^2 = \|Uy_t - Uy_s\|^2 . \tag{25}$$

The matrix M (or equivalently U) must be designed to better highlight the change points to detect [87].
Using a Mahalanobis-type metric only involves linear treatments of the samples. One way to introduce non-linearity is to combine it with a kernel-based transformation. Practically, for a given kernel function $k(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}$ and its associated mapping $\phi(\cdot) : \mathbb{R}^d \mapsto \mathcal{H}$ onto the RKHS $\mathcal{H}$, the Mahalanobis-type metric $\|\cdot\|_{\mathcal{H},M}$ in the feature space $\mathcal{H}$ is defined by

$$\|\phi(y_s) - \phi(y_t)\|_{\mathcal{H},M}^2 = (\phi(y_s) - \phi(y_t))' M (\phi(y_s) - \phi(y_t)) \tag{26}$$

where M is a (possibly infinite dimensional) symmetric positive semi-definite matrix defined on $\mathcal{H}$. The resulting cost function is given by

$$c_{\mathcal{H},M}(y_{a..b}) := \sum_{t=a+1}^{b} \|y_t - \bar{y}_{a..b}\|_{\mathcal{H},M}^2 \tag{27}$$

3.5. Summary of cost functions

Name	$c(y_{a..b})$	Parameters
$c_{\text{i.i.d.}}$ (8)	$-\sup_\theta \sum_{t=a+1}^{b} \log f(y_t\|\theta)$	θ: changing parameter; density function: $f(\cdot\|\theta)$
c_{L_2} (9)	$\sum_{t=a+1}^{b} \|y_t - \bar{y}_{a..b}\|_2^2$	$\bar{y}_{a..b}$: empirical mean of $y_{a..b}$
c_Σ (10)	$\log\det \widehat{\Sigma}_{a..b}$	$\widehat{\Sigma}_{a..b}$: empirical covariance of $y_{a..b}$
c_{Poisson} (11)	$-(b-a)\bar{y}_{a..b} \log \bar{y}_{a..b}$	$\bar{y}_{a..b}$: empirical mean of $y_{a..b}$
c_{linear} (13)	$\min_{u\in\mathbb{R}^p, v\in\mathbb{R}^q} \sum_{t=a+1}^{b} (y_t - x_t'u - z_t'v)^2$	$x_t \in \mathbb{R}^p$, $z_t \in \mathbb{R}^q$: covariates
c_{linear,L_1} (14)	$\min_{u\in\mathbb{R}^p, v\in\mathbb{R}^q} \sum_{t=a+1}^{b} \|y_t - x_t'u - z_t'v\|$	$x_t \in \mathbb{R}^p$, $z_t \in \mathbb{R}^q$: covariates
c_{AR} (Example 4)	$\min_{u\in\mathbb{R}^p} \sum_{t=a+1}^{b} (y_t - x_t'u)^2$	$x_t = [y_{t-1}, y_{t-2}, \ldots, y_{t-p}]$: lagged samples
c_{kernel} (17)	$\sum_{t=a+1}^{b} k(y_t, y_t) - \frac{1}{b-a}\sum_{s,t=a+1}^{b} k(y_s, y_t)$	$k(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}$: kernel function
c_{rbf} (19)	$(b-a) - \frac{1}{b-a}\sum_{s,t=a+1}^{b} \exp(-\gamma \|y_s - y_t\|^2)$	$\gamma > 0$: bandwidth parameter
c_M (23)	$\sum_{t=a+1}^{b} \|y_t - \bar{y}_{a..b}\|_M^2$	$M \in \mathbb{R}^{d\times d}$: positive semi-definite matrix
$c_{\mathcal{H},M}$ (27)	$\sum_{t=a+1}^{b} \|y_t - \bar{y}_{a..b}\|_{\mathcal{H},M}^2$	M: positive semi-definite matrix (in the feature space $\mathcal{H}$)

Table 1: Summary of cost functions

A summary of the described cost functions is displayed in Table 1.

4. Search methods with a fixed K

We now present the algorithmic methods to solve (exactly or approximately) the discrete change point detection optimization problem (5) when the number of change points to detect is fixed to K. Denote by $\hat{\mathbf{t}}_K(y)$ the minimizer of the contrast function $V(\cdot)$ that contains K change points. Namely,

$$\hat{\mathbf{t}}_K(y) := \underset{|\mathbf{t}|=K}{\arg\min}\, V(\mathbf{t}, y). \tag{28}$$

The optimal change point set $\hat{\mathbf{t}}_K(y)$ is denoted $\hat{\mathbf{t}}_K$ when it is obvious from the context that it refers to the signal y.

4.1. Optimal detection

Change point detection with a known number of changes K (5) is a discrete optimization problem over the set of indexes of $\{1, \dots, T\}$. The set of all possible combinations of indexes has a cardinal of $\binom{T-1}{K}$: this makes an exhaustive search impossible. However, a more efficient method based on dynamic programming is available. Historically, it was introduced for a non-related problem [23] and later applied to change point detection [18, 58, 75]. In the following, the dynamic programming change point detection method is referred to as Opt.

Practically, dynamic programming relies on the additive nature of the objective function $V(\cdot)$. Roughly, it consists in recursively solving sub-problems and rests on the following observation:

$$\begin{aligned}
\min_{|\mathbf{t}|=K} V(\mathbf{t}, y = y_{0..T}) &= \min_{0=t_0<t_1<\cdots<t_K<t_{K+1}=T} \sum_{k=0}^{K} c(y_{t_k..t_{k+1}}) \\
&= \min_{t \leq T-K} \left[c(y_{0..t}) \;+\; \min_{t=t_0<t_1<\cdots<t_{K-1}<t_K=T} \sum_{k=0}^{K-1} c(y_{t_k..t_{k+1}}) \right] \\
&= \min_{t \leq T-K} \left[c(y_{0..t}) \;+\; \min_{|\mathbf{t}|=K-1} V(\mathbf{t}, y_{t..T}) \right]
\end{aligned} \tag{29}$$

Intuitively, Equation 29 means that the first change point of the optimal segmentation is easily computed if the optimal partitions with $K-1$ elements of all sub-signals $\{y_s\}_t^T$ are known. The complete partition is then computed by recursively applying this observation. Note that Opt solves exactly the change point detection problem (5) (when K is known).

When combined with the quadratic error cost function (Example 1), the complexity of Opt is $\mathcal{O}(KT^2)$ [18, 76]. More complex cost functions increase computational complexity. The detailed algorithm can be found in Section 9 (Algorithm 1).

The optimal change point detection method Opt is applied on biological sequences (EEG recordings in [88, 90], tree growth monitoring [58] for instance) and financial time-series [89]. Since its algorithmic complexity is quadratic, signals on which Opt is applied are relatively short with around a hundred samples.

4.1.1. Extensions

Several extensions have been proposed in the literature to reduce the computational burden of Opt. The proposed methods still find the optimal set of change points $\hat{\mathbf{t}}_K$.

- The "pruned optimal dynamic programming" procedure [118] is an extension of Opt that relies on a pruning rule to discard indexes that can never be change points. Thanks to this trick, the set of potential change point indexes is reduced. All described cost functions can be plugged into this method.
  As a result, longer signals can be handled, for instance long array-based DNA copy number data (up to 10^6 samples, with the quadratic error cost function) [118]. However, worst case complexity remains of the order of $\mathcal{O}(KT^2)$.

- The "forward dynamic programming" is described in [58]. While Opt only computes the optimal partition which minimizes the sum of costs $V(\cdot)$, the "forward dynamic programming" computes the top L ($L \geq 1$) most probable partitions (with lowest sum of costs). The algorithm is detailed for cost functions of the form (8), which detect change in the distribution of piecewise i.i.d. random variables. The computational complexity is $\mathcal{O}(LKT^2)$ where L is the number of computed partitions.
  This procedure is designed as a diagnostic tool: change points present in many of the top partitions are considered very likely, while change points present in only a few of the top partitions might not be as relevant. Thanks to "forward dynamic programming", insignificant change points are trimmed and overestimation of the number of change point is corrected [58], at the expense of a higher computational burden. It is applied to tree growth monitoring time series [58] that are relatively short with around a hundred samples.

4.2. Window sliding

Window-sliding algorithm (referred to as Win in the following) is a fast approximate alternative to Opt. This scheme relies on *single* change point detection procedures and extends it to find multiple change points.
Algorithmically, two adjacent windows slide along the signal. The discrepancy between the first window and the second window is computed. For a given cost function $c(\cdot)$, the discrepancy between two sub-signals is of the form

$$d(y_{a..t}, y_{t..b}) = c(y_{a..b}) - c(y_{a..t}) - c(y_{t..b}) \quad (1 \leq a < t < b \leq T). \tag{30}$$

When the two windows cover dissimilar segments, the discrepancy reaches large values and a change point is detected. Methods from the literature also use discrepancy measures that stem from a two-sample test setting (see Remark 5). The two standpoints are often equivalent: for instance, using c_{L_2}, $c_{\text{i.i.d.}}$ or c_{kernel} is respectively equivalent to applying a Student t-test, a generalized likelihood ratio (GLR) test and a kernel Maximum Mean Discrepancy (MMD) test [56, 57]. Certain homogeneity tests however are not derived from a well-designed cost function. In the context of *single* change point detection, examples of such tests include rank-based procedures [38, 94, 97], kernel Fisher discriminants [62, 63, 103], density ratio [74], etc.

Remark 5 (Two-sample test)**.** *A two-sample test (or homogeneity test) is a statistical hypothesis testing procedure designed to assess whether two populations of samples are identical in distribution. Formally, consider two sets of i.i.d. $\mathbb{R}^d$-valued random samples $\{x_t\}_t$ and $\{z_t\}_t$. Denote by $\mathbb{P}_x$ the distribution function of the x_t and by $\mathbb{P}_z$, the distribution function of the z_t. A two-sample test procedure compares the two following hypotheses:*

$$\begin{aligned} H_0 &: \quad \mathbb{P}_x = \mathbb{P}_z \\ H_1 &: \quad \mathbb{P}_x \neq \mathbb{P}_z. \end{aligned} \tag{31}$$

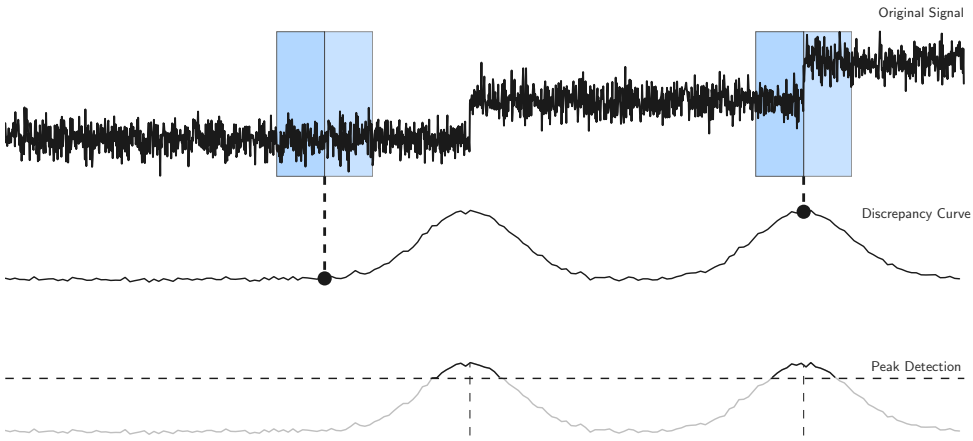

Figure 2: Schematic view of Win

A general approach is to consider a probability (pseudo)-metric $d(\cdot, \cdot)$ on the space of probability distributions on $\mathbb{R}^d$. Well-known examples of such a metric include the Kullback-Leibler divergence, the Kolmogorov-Smirnov distance, the Maximum Mean Discrepancy (MMD), etc. Observe that, under the null hypothesis, $d(\mathbb{P}_x, \mathbb{P}_z) = 0$. The testing procedure consists in computing the empirical estimates $\widehat{\mathbb{P}}_x$ and $\widehat{\mathbb{P}}_z$ and rejecting H_0 for "large" values of the statistics $d(\widehat{\mathbb{P}}_x, \widehat{\mathbb{P}}_z)$.
This is a general formulation which is difficult because it relies on a consistent estimation of arbitrary distributions from a finite number of samples. Additional assumptions can be made on the distribution functions that yield simpler problems: for instance, Gaussian assumption [22, 32, 33], exponential family assumption [50, 114], etc. In the more complex non-parametric setting, the distributions are only assumed to be continuous. They are not directly estimated; instead, the statistics $d(\widehat{\mathbb{P}}_x, \widehat{\mathbb{P}}_z)$ is computed [38, 57, 63, 95].
In the context of single *change point detection, the two-sample test setting is adapted to assess whether a distribution change has occurred at some instant in the input signal. Practically, for a given index t, the homogeneity test is performed on the two populations $\{y_s\}_{s \leq t}$ and $\{y_s\}_{s > t}$. The estimated change point location is given by*

$$\hat{t} = \arg\max_t \; d(\widehat{\mathbb{P}}_{\bullet \leq t}, \widehat{\mathbb{P}}_{\bullet > t}) \tag{32}$$

where $\widehat{\mathbb{P}}_{\bullet \leq t}$ and $\widehat{\mathbb{P}}_{\bullet > t}$ are the empirical distributions of respectively $\{y_s\}_{s \leq t}$ and $\{y_s\}_{s > t}$.

In the offline setting, the complete discrepancy curve is computed and a peak search procedure is performed to find change point indexes. The Win algorithm is detailed in Algorithm 4; a schematic view is displayed on Figure 2.
The main benefits of Win are its low complexity (linear when used with c_{L_2}) and ease of implementation. In addition, any *single* change point detection method can be plugged into this scheme. An extensive literature is available in which asymptotic distributions of the discrepancy (for several cost functions) [32, 57, 63, 94, 97]; this is useful to find suitable thresholds in the peak search procedure. Nevertheless, Win is often unstable because the *single* change point

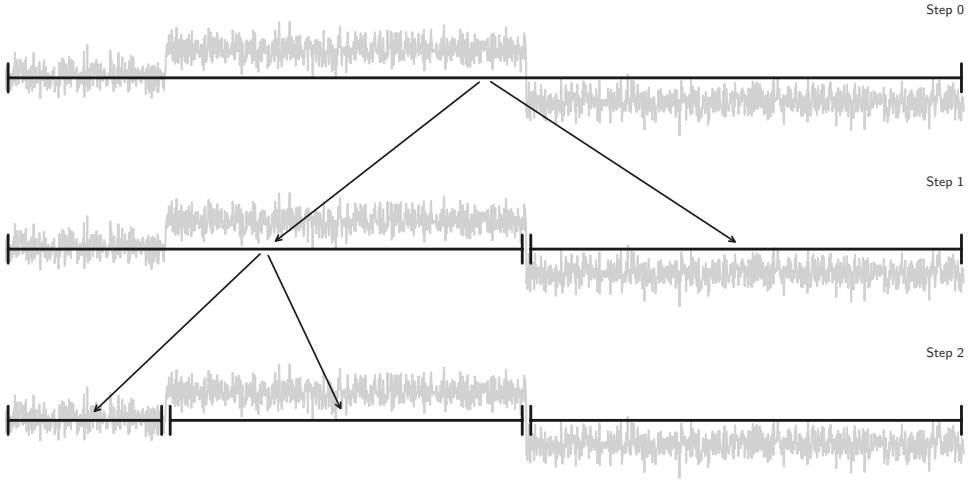

Figure 3: Schematic example of BinSeg

detection is performed on small regions (the window), lowering its statistical power [47, 59, 73]. Win has been applied in numerous settings with many different cost functions. For instance, it has been applied on biological signals [28, 47, 63, 73, 131], on network data [94, 97], on speech time series [1, 42, 63] and on financial time series [22, 34, 78].

4.3. Binary segmentation

Binary segmentation (referred to as BinSeg in the following) is a well-known fast and approximate alternative to Opt [123]. In the context of change point detection, BinSeg is one of the most used methods because of its conceptual simplicity and ease of implementation [33, 81, 106].

BinSeg is a greedy sequential algorithm: at each iteration $k \geq 1$, detection of a single change point is performed and an estimate $\hat{t}^{(k)}$ is produced. (In the following, the superscript $\cdot^{(k)}$ refers to k-th step of a sequential algorithm.)

The first change point estimate $\hat{t}^{(1)}$ is given by

$$\hat{t}^{(1)} := \underset{1 \leq t < T-1}{\arg\min} \underbrace{c(y_{0..t}) + c(y_{t..T})}_{V(\mathbf{t} = \{t\})}. \tag{33}$$

This operation is "greedy", in the sense that it searches the change point that lowers the most the sum of costs. The signal is then split in two at the position of $\hat{t}^{(1)}$; the same operation is repeated on the resulting sub-signals until no further change point is detected. Formally, after k steps of BinSeg, the next change point estimate $\hat{t}^{(k+1)}$ belongs to one of the $k+1$ sub-segments of the original signal and given by

$$\hat{t}^{(k+1)} := \underset{t \notin \{\hat{t}^{(1)}, \ldots, \hat{t}^{(k)}\}}{\arg\min} V(\{\hat{t}^{(1)}, \ldots, \hat{t}^{(k)}\} \cup \{t\}). \tag{34}$$

Since the number of change points K is known, BinSeg needs K steps to estimate the indexes. The greedy detection step (33) can be formulated as a two-sample testing procedure (see Remark 5). The two standpoints are often equivalent but certain homogeneity tests however are not derived from a well-designed cost function (see comments in Section 4.2). A schematic view of the algorithm is displayed on Figure 3; an implementation can be found in Section 9 (Algorithm 2). The complexity of BinSeg is $\mathcal{O}(T \log T)$ for the quadratic error loss c_{L_2} (9).
Theoretically, BinSeg is proven to produce asymptotically consistent estimates under the mean-shift model (Example 1) with the quadratic error c_{L_2} (9) [9, 130]. A more recent result states that BinSeg is only consistent when the minimum spacing between any two adjacent change-points is of order greater than $T^{3/4}$ [52].
In general, BinSeg's output is only an approximation of the optimal estimates (the output of Opt). As argued in [9, 81], the issue is that the estimated change points $\hat{t}^{(k)}$ are not estimated from homogeneous segments and each estimate depends on the previous ones. Change points that are close are imprecisely detected especially [72].
Applications of BinSeg range from financial time series [9, 32, 33, 52, 92] to context recognition for mobile devices [67] and array-based DNA copy number data [106, 112].

4.3.1. Extensions

Several extensions of BinSeg have been proposed to improve detection accuracy. Indeed, small segments tend to be overlooked by BinSeg [9, 106]. However, these adaptation often come at the expense of implementation ease.

- Circular binary segmentation [106] is a well-known extension of BinSeg. This method is also a sequential detection algorithm that splits the original at each step. Instead of searching for a single change point in each sub-signal, circular binary segmentation searches two change points. Within each treated sub-segment, it assumes a so-called "epidemic change model": the parameter of interest shifts from one value to another at the first change point and returns to the original value at the second change point. The algorithm is dubbed "circular" because, under this model, the sub-segment has its two ends (figuratively) joining to form a circle.
  Practically, this method is used to detect changes in the mean of array-based DNA copy number data [86, 106, 132] (with the quadratic error cost function 9). A faster version of the original algorithm is described in [128].

- Another extension of BinSeg is the wild binary segmentation algorithm [52]. In a nutshell, a single point detection is performed on multiple intervals with start and end points that are drawn uniformly. Small segments are likely to contain at most one change but have lower statistical power, while the opposite is true for long segments. After a proper weighting of the change score to account for the differences on sub-signals' length, the algorithm returns the most "pronounced" ones, i.e. those that lower the most the sum of costs. An important parameter of this method is the number of random sub-segments to draw; its calibration is discussed Wild binary search is used to detect mean-shifts of univariate piecewise constant signals (up to 2000 samples) [52].

4.4. Bottom-up segmentation

Bottom-up segmentation is the natural counterpart of BinSeg. It is referred to as BotUp in the following. This algorithm is also a sequential segmentation method that approximates the

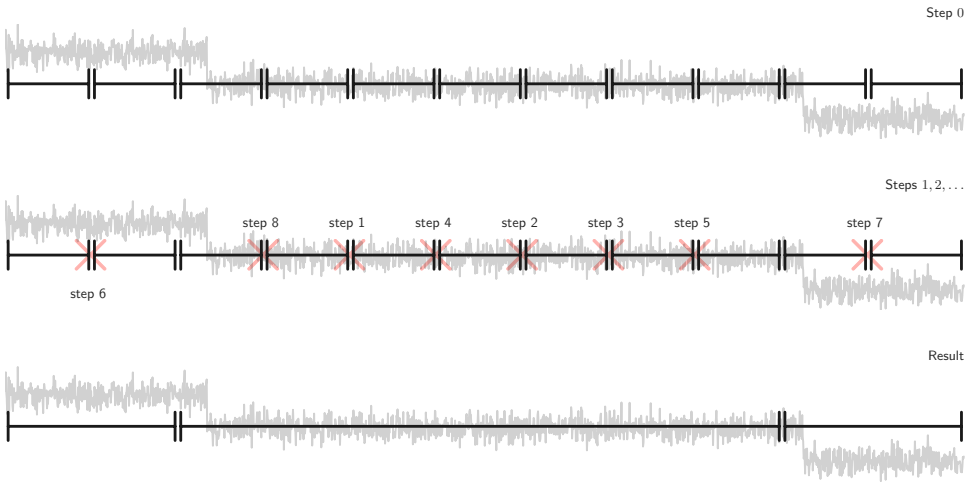

Figure 4: Schematic view of BotUp

exact change point detection [34, 51, 77].

Contrary to BinSeg, BotUp starts by splitting the original signal in many small sub-signals and sequentially merges them until there remain only K change points. At every step, all potential change points (indexes separating adjacent sub-segments) are ranked by the discrepancy between the segments they separate. Change points with the lowest discrepancy are then deleted, meaning that the segments they separate are merged. Recall that (see Section 4.2 and Equation 30), for a given cost function $c(\cdot)$, the discrepancy between two sub-signals is of the form

$$d(y_{a..t}, y_{t..b}) = c(y_{a..b}) - c(y_{a..t}) - c(y_{t..b}) \quad (1 \leq a < t < b \leq T). \tag{35}$$

It is related to the decrease in the sum of costs $V(\cdot)$ the change point is responsible for. As is the case for BinSeg, two-sample tests (Remark 5) can also be used to provide a similar measure of discrepancy (Equation 32).

BotUp is often dubbed a "generous" method, by opposition to BinSeg, which is "greedy" [77]. A schematic view of the algorithm is displayed on Figure 4; an implementation can be found in Section 9 (Algorithm 3).

Its benefits are its linear computational complexity (in the number of samples T, for the quadratic error cost) and conceptual simplicity. However, several downsides exist. First, if a true change point does not belong to the original set of indexes, BotUp never considers it. Moreover, in the first iterations, the merging procedure (35) can be unstable because it is performed on small segments, for which statistical significance is smaller. In the literature, BotUp is somewhat less studied than its counterpart, BinSeg: no theoretical convergence study is available. It has been applied on speech time series to detect mean and scale shifts [34]. Besides, the authors of [77] have found that BotUp outperforms BinSeg on ten different data sets such as physiological signals (ECG), financial time-series (exchange rate), industrial monitoring (water levels), etc.

5. Search methods with a penalty

We now present the strategies to estimate the number and locations of change points when their number K is unknown. In particular, we provide insights on how to choose the penalty function $\text{pen}(\cdot)$ and the penalty level β, and then on how to solve the related optimization problem (6).

5.1. l_0 penalty

The l_0 penalty (or linear penalty) is arguably the most popular choice of penalty [81]. First introduced in [68] and later in [134, 135], it was then studied under more general assumptions [24–26].
Formally, the l_0 penalty is denoted by pen_{l_0} and is given by

$$\text{pen}_{l_0}(\mathbf{t}) := \beta|\mathbf{t}| \tag{36}$$

where $\beta > 0$ controls the trade-off between complexity and goodness-of-fit. Intuitively, it balances the decrease of the sum of costs $V(\cdot)$ that arises when more and more change points are allowed. This formulation generalizes several popular criterion from the literature such as BIC and AIC (see Example 5).

Example 5 (Bayesian Information Criterion (BIC) and Akaike Information Criterion (AIC))**.** *The well-known BIC and AIC criteria are special cases of the l_0 penalty (36). Assume a piecewise i.i.d. model for the signal y (see Section 3.1). Upon setting $c = c_{\text{i.i.d.}}$, the constrained likelihood which leads to BIC can be formulated for a given set of change points $\mathbf{t}$ as follows:*

$$2V(\mathbf{t}) + p \log T|\mathbf{t}| \tag{37}$$

where p is the dimension of the parameter space ($p = 1$ for a univariate change in mean, $p = 2$ for a univariate change in mean and scale,...). BIC is therefore equivalent to the linearly penalized change point detection (36) upon setting $\beta = p/2 \log T$:

$$\textit{pen}_{\textit{BIC}}(\mathbf{t}) := \frac{p}{2} \log T \ |\mathbf{t}| \tag{38}$$

Similarly, under the mean-shift model (with fixed variance), with $c = c_{L_2}$, the BIC penalty [121, 134] is

$$\textit{pen}_{\textit{BIC},L_2}(\mathbf{t}) := \sigma^2 \log T \ |\mathbf{t}|. \tag{39}$$

The same holds for AIC, for which

$$\textit{pen}_{\textit{AIC},L_2}(\mathbf{t}) := \sigma^2 \ |\mathbf{t}|. \tag{40}$$

AIC is a generalization of the also well-known Mallows' C_p [101].

Theoretically, consistency of the change point estimates can be proven in certain situations. The most studied model is the mean-shift model for which the estimated number of change points and change point fractions converge in probability asymptotically under various assumptions:

- in the noiseless setting, if the penalty value β converges at an appropriate rate to zero [26],
- if the noise is Gaussian white noise and $\text{pen} = \text{pen}_{\text{BIC},L_2}$ [134],

- if the noise is a second order stationary process (with appropriately decreasing autocorrelation function) and the penalty level β is slowly diverging to infinity (slower than T) [90, 91].

A recent result states the consistency of the linearly penalized change point estimates for $c = c_{\text{kernel}}$ for penalty values β converging to 0 at the speed of $1/T$ [55]. Theoretical results for other cost functions (most notably $c_{\text{i.i.d.}}$) only deals with the convergence of the sum of costs to its true value [134, 135]; no results are available about the consistency of the change point estimates.

5.1.1. Implementation

A naive heuristics to solve the linearly penalized change point detection is to apply Opt for $K = 1, \dots, K_{\max}$ for a sufficiently large $K_{\max}$. Then choose among the computed partitions the one that minimizes the penalized problem. This approach would prove computational cumbersome because of the quadratic complexity of the resolution method Opt. Fortunately a faster method exists with far lower complexity.

The algorithm Pelt (Pruned Exact Linear Time) was introduced to find the exact solution of (6) with pen $= \text{pen}_{l_0}$ [81]. This approach considers each sample sequentially and, thanks to an explicit pruning rule, may or may not discard it from the set of potential change points. This results in a considerable speed-up: under the assumption that regime lengths are randomly drawn from a uniform distribution, the complexity of Pelt is of the order $\mathcal{O}(T)$. However, as argued in [52], this set-up is unrealistic as it produces shorts segments. Nevertheless, in practice, this method is orders of magnitude faster than the naive heuristics previously described [69, 81]. The detailed algorithm can be found in Section 9 (Algorithm 5).

An extension of Pelt is described in [65] to solve the linearly penalized change point detection for a range of penalty values $[\beta_{\min}, \beta_{\max}]$.

When using Pelt, the only parameter to calibrate is the penalty level β. Small values encourage partitions with many change points, while large values make Pelt sensitive only to significant changes. Theoretical results (in an asymptotic setting) suggest that β should at least depend on the number of samples and the dimension of the signal. BIC, AIC and Mallow's C_p (see Example 5) are statistical procedures to set the value of the penalty. However, in practice, numerical experiments have demonstrated that they significantly overestimate the number of change points [58, 69, 135]

There are also several heuristics in the literature. In [92, 112] Opt is applied several times on one training signal to find a suitable β that can be used to segment new signals, An adapted cross-validation procedure is described in [3] and [25] proposes another method based on the slope heuristics.

A different route is taken by [69]. They propose a supervised procedure: the chosen penalty value is the one that minimizes the segmentation error on an annotated set of signals. However, it requires to perform several times the optimal change point detection Opt on each signal, which proves to be computationally heavy.

The main application of Pelt is DNA sequence data [69, 70, 135, 137] (with the cost function c_{L_2}. It has also been applied to financial time-series [92] and oceanographic data [81], where the exact penalized change point has be shown to outperform faster heuristics like BinSeg.

5.2. l_1 penalty

To further lower the computational cost of change point detection with a linear penalty, an alternative formulation has been proposed in which the l_0 penalty is relaxed to a l_1 penalty [60, 129]. The same reasoning is central to many developments in machine learning, for instance sparse regression, compressive sensing, sparse PCA, dictionary learning [64]. In numerical analysis and image denoising, this penalty is also known as the total variation regularizer [61, 122, 129].

This strategy has been introduced to detect mean-shifts in piecewise constant signals with Gaussian noise (see Example 1); the related function is c_{L_2}. Under this setting, the change point optimization problem (6) is rewritten as follows:

$$\min_{\mathbf{t}} \; V(\mathbf{t}) \; + \beta \sum_{k=1}^{|\mathbf{t}|} \left\| \bar{y}_{t_{k-1}..t_k} - \bar{y}_{t_k..t_{k+1}} \right\|_1 \tag{41}$$

where $\bar{y}_{t_{k-1}..t_k}$ is the empirical mean of sub-signal $y_{t_{k-1}..t_k}$. The l_1 is then given by

$$\text{pen}_{l_1}(\mathbf{t}) := \beta \sum_{k=1}^{|\mathbf{t}|} \left\| \bar{y}_{t_{k-1}..t_k} - \bar{y}_{t_k..t_{k+1}} \right\|_1 . \tag{42}$$

In [60, 129], the authors show that it is in fact equivalent to a convex optimization problem (known in other contexts as total variation denoising). Introduce the matrices $Y := [y_1, \dots, y_T]' \in \mathbb{R}^{T \times d}$ and $S \in \mathbb{R}^{T \times T-1}$ given by

$$S_{ij} := \begin{cases} 1 \text{ if } i > j, \\ 0 \text{ if } i \leq j. \end{cases} \tag{43}$$

Let $\bar{Y}$ and $\bar{S}$ denote the matrices obtained from Y and S by centering each column. Then the change point detection problem (41) with l_1 penalty is equivalent to

$$\min_{\Delta \in \mathbb{R}^{(T-1) \times d}} \left\| \bar{Y} - \bar{S}\Delta \right\|^2 \; + \; \beta \underbrace{\sum_{t=1}^{T-1} \|\Delta_{t,\bullet}\|_1}_{\|\Delta\|_{1,1}} . \tag{44}$$

The estimated change point indexes are the support rows (i.e. the rows that are non-zero) of Δ. The matrix Δ is the "jump matrix": it contains the locations and the amplitudes of the mean-shifts of the signal $\bar{S}\Delta$. This problem (44) is an instance of the well-known Lasso (for "least absolute shrinkage and selection operator") regression [64, 127] with design matrix $\bar{S}$. Intuitively, the l_1 penalty shrinks many coefficients to zero, which in this context means that it favours piecewise constant approximations of y. The higher the penalty value β is, the less change points are allowed.

Algorithmically, the optimization is carried out with the least-angle regression (Lars) [45]. This method has a complexity of $\mathcal{O}(T \log T)$, where an upper bound on the number of change points is known.

From a theoretical standpoint, the estimated change point fractions are asymptotically consistent [61]. This result is proven for an appropriately converging sequence of penalty values β.

In particular, the number of change points is also correctly estimated (asymptotically). This consistency property is obtained even though the design matrix $\bar{S}$ does not verify classical assumptions from the Lasso regression framework (such as the irrepresentable condition) [61].
In practice, there is no method to calibrate the regularization parameter β. Only a "trial and error" or cross-validation procedures are used in the literature [2, 129].
The l_1 penalized change point detection has been extended to deal with changes in autoregressive coefficients of a piecewise stationary signal [2] (the related cost function is c_{AR}, see Example 4). In [129] the l_1 regularization (44) is replaced with a l_2 penalty (the so-called Tikhonov regularization). The consistent of the *first* change point index is proven as the dimension d of the signal diverges to infinity (for a fixed number of samples T).
To summarize, l_1 penalty was introduced to detect mean-shifts when the number of change points is unknown. It was later extended to deal with abruptly changing autoregressive coefficients. Main applications include array-based DNA copy number data [70, 129] and speech signals [2].

5.3. Other penalties

Several other penalty functions an be found in the literature. In general, they stem from theoretical considerations, under the mean-shift model with the cost function c_{L_2}, and are complex to use in practice.
The modified BIC criterion [137] is a penalty which depends not only on the number of change points but also on the lengths of their distances from each other:

$$\text{pen}_{\text{mBIC}}(\mathbf{t}) := 3|\mathbf{t}| \log T + \sum_{k=0}^{|\mathbf{t}|+1} \log(\frac{t_{k+1} - t_k}{T}). \tag{45}$$

The second term of pen_{mBIC} favours evenly spaced change points; it reaches its maximum when indexes are close to each other, for instance $t_1 = 1, t_2 = 2, \dots$ In practice, the associated change point detection problem is not tractable and therefore this penalty is approximated by the linear penalty pen_{l_0}.
In [93], a model selection procedure for univariate piecewise constant signals leads to the following function:

$$\text{pen}_{\text{Leb}}(\mathbf{t}) := \frac{|\mathbf{t}| + 1}{T} \sigma^2 (a_1 \log \frac{|\mathbf{t}| + 1}{T} + a_2) \tag{46}$$

where a_1 and a_2 are real constants to calibrate and σ^2 is the noise variance. To perform the associated change point detection task, one must find the optimal partitions with fixed size $K = 1, 2, \dots, K_{\max}$ for a sufficiently large $K_{\max}$, then find among the computed partitions the one that minimizes the penalized sum of costs. To tune the constants a_1 and a_2, the authors propose to use cross-validation.
In the situation when the penalty $\text{pen}(\cdot)$ only depends on the cardinal of $\mathbf{t}$ and is concave, the penalty can be replaced by a linear penalty function pen_{l_0} with a suitable penalty value β [81]. As a consequence, for this class of penalty functions, change point detection can be performed using Pelt.

6. Bayesian methods

Bayesian methods are an important of the change point detection literature. Those methods can accommodate prior knowledge on the change point location distribution. They are widely

used in certain areas such as speech recognition [116], brain imaging [114], video segmentation [109] and bioinformatics [21, 44, 48].
Bayesian methods generally assume an unobserved discrete state variable s_t ($t = 1, \dots, T$) which controls the model producing the data.. Change points are indexes where the hidden variable s_t goes from one state to another. The signal y is assumed to be piecewise i.i.d. such that

$$y_t \sim f(\cdot|\theta_{s_t}) \tag{47}$$

where $f(\cdot|\theta)$ is the probability distribution of an observation characterized by the parameter θ, θ_i are emission parameters for an observation in state i. The joint probability of the observations is given by

$$\mathbb{P}(y, [s_1, \dots, s_T]) = \prod_{t=1}^{T} f(y_t|\theta_{s_t}). \tag{48}$$

Bayesian algorithms differ from each other by the prior that is assumed on the state variable s_t distribution and the method that is used to compute the state sequence which best "explains" the observations.

A well studied Bayesian method is the Hidden Markov Model (HMM). General overviews can be found in [29, 116]. The state variable is assumed to be a Markov chain, meaning that the value of s_t only depends on its previous value s_{t-1}. This probabilistic description is entirely summarized by a transition matrix A such that

$$A_{ij} = \mathbb{P}(s_t = j|s_{t-1} = i). \tag{49}$$

The HMM is defined by three parameters: the transition matrix A, the emission probabilities $f(\cdot|\theta_i)$ and the initial state probabilities $\mathbb{P}(s_1 = \theta_i)$.
Change point detection in this context amounts to finding the maximum a posteriori (MAP) state sequence $\hat{s}_1, \dots, \hat{s}_T$. Computations are usually performed using the Viterbi algorithm or by random sampling methods (such as Monte Carlo Markov Chain, MCMC [46, 48]) if prior distributions on the parameters are provided.
Benefits of HMM include state-of-the-art performances on certain applications such as speech processing [29, 104] and the ability to model any piecewise i.i.d. signal. They are also well-understood methods with many available implementations. However, HMM rely on several assumptions that are not verified in many contexts: observations are assumed independent, the emission distribution is parametric, usually a Gaussian mixture, the Markov assumption is not adapted to long segments, for instance. Several related algorithms are developed to overcome those limitations.
Switching linear dynamic models have greater descriptive power than independent emissions [109]. They are applied to the complex task of visual tracking. However the calibration step is significantly more burdensome as there many parameters to tune.
The Markov assumption can be relaxed and semi-Markov models are considered [102]. Semi-Markov processes keep track of the length of the current segment and are more suited to model state variables with few change points.
Developed by [82] with strong connections to [36], Dirichlet process hidden Markov models are well adapted to perform change point detection. The number of states is implicitly computed and sampling from the posteriori distribution is made computationally easy.

7. Summary Table

The literature review is summarized in Table 2.
When applicable, each publication is associated with a search method (such as Opt, Pelt, BinSeg or Win); this is a rough categorization rather than an exact implementation.
We provide a guide to assess computational complexity. Quadratic methods are the slowest and have only one star while linear methods are given three stars.
Algorithms for which the number of change points is an explicit input parameter work under the "known K" assumption. Algorithms that can be used even if the number of change points is unknown work under the "unknown K" assumption. (Certain methods can accommodate both situations.)
Some methods are implemented and available online. A detailed summary of libraries can be found in Table 3.

Publication	Search method	Cost function	Known K		Scalability (w.r.t. T)	Package	Additional information
			Yes	No			
Sen and Srivastava [123], Vostrikova [130]	BinSeg	c_{L_2}	✓	-	★★★	✓	
Yao [134]	Pelt	c_{L_2}	-	✓	★☆☆	-	Bayesian information criterion (BIC)
Cappé et al. [29], Rabiner [116]	Bayesian	$c_{\text{i.i.d.}}$	-	✓	★★☆	-	HMM
Basseville and Nikiforov [22]	Opt	$c_{\text{i.i.d.}}$, c_{L_2}	-	-	★★★	-	single change point
Bai [6], Bai and Perron [18]	Opt	c_{linear, L_2}	-	-	★★☆	-	single change point
Bai [7]	Opt	c_{linear, L_1}	-	-	★★☆	-	single change point
Lavielle [88]	Opt	c_{AR}	✓	-	★☆☆	-	
Chib [36]	Bayesian	$c_{\text{i.i.d.}}$	-	✓	★★☆	-	Dirichlet process
Bai [13]	Opt	c_{AR}	✓	-	★☆☆	-	
Birgé and Massart [24, 25]	Opt	c_{L_2}	-	✓	★☆☆	-	model selection
Bai and Perron [16]	Opt	c_{L_2}	✓	-	★☆☆	-	
Olshen et al. [106], Venkatraman and Olshen [128]	BinSeg	c_{L_2}	✓	✓	★★★	✓	
Lebarbier [93]	Opt	c_{L_2}	-	✓	★☆☆	-	model selection
Desobry et al. [42]	Win	c_{kernel}	-	✓	★★★	-	dissimilarity measure (one-class SVM), see Remark 5
Harchaoui and Cappé [59]	Opt	c_{kernel}, c_{rbf}	✓	-	★☆☆	-	
Zhang and Siegmund [137]	Pelt	c_{L_2}	-	✓	★★☆	-	modified BIC
Harchaoui et al. [63]	Win	-	✓	✓	★★★	-	dissimilarity measure (Fisher discriminant), see Remark 5
Kawahara [74]	Win	-	-	✓	★★★	-	dissimilarity measure (density ratio), see Remark 5
Lévy-Leduc and Roueff [94], Lung-Yut-Fong et al. [97]	Win	-	✓	✓	★★★	-	dissimilarity measure (rank-based), see Remark 5
Bai [14]	Opt	c_{L_2}, c_{Σ}	-	-	★★☆	-	single change point
Harchaoui and Lévy-Leduc [60]	Lasso	c_{L_2}	-	✓	★★★	-	total variation regression (pen_{l_1})
Vert and Bleakley [129]	Lasso	c_{L_2}	-	✓	★★★	-	Tikhonov regularization
Killick et al. [81]	Pelt	any $c(\cdot)$	-	✓	★★☆	✓	
Arlot et al. [4]	Opt	c_{kernel}, c_{rbf}	✓	✓	★☆☆	-	
Angelosante and Giannakis [2]	Lasso	c_{AR}	-	✓	★★★	-	Tikhonov regularization
Hocking et al. [69]	Pelt	c_{L_2}	-	✓	★★☆	-	supervised method to learn a penalty level (pen_{l_0})
Fryzlewicz [52]	BinSeg	c_{L_2}	✓	✓	★★★	✓	univariate signal
Lajugie et al. [87]	Opt	c_M	✓	-	★☆☆	-	supervised method to learn a suitable metric
Frick et al. [50]	BinSeg	$c_{\text{i.i.d.}}$	✓	✓	★★★	✓	exponential distributions family
Ko et al. [82]	Bayesian	$c_{\text{i.i.d.}}$	-	✓	★★☆	-	Dirichlet process
Garreau and Arlot [55]	Pelt	c_{kernel}, c_{rbf}	✓	✓	★☆☆	-	
Haynes et al. [65]	Pelt	any $c(\cdot)$	-	✓	★★☆	-	
Chakar et al. [31]	Pelt	c_{AR}	✓	✓	★☆☆	✓	

Table 2: Summary table of literature review.

8. Conclusion

In this work, we have reviewed numerous methods to perform change point detection, organized within a common framework. Precisely, all methods are described as a collection of three elements: a cost function, a search method and a constraint on the number of changes to detect. For a given method, we detail the assumed signal model, the associated algorithm, theoretical guarantees (if any) and the application domain.

As was shown above, contributions to this large area of research are diverse: they encompass many different models and operational constraints such precision, complexity, etc. The formal framework for change point detection was introduced to give sens to this significant body of work. This approach is intended to facilitate prototyping of change point detection methods: for a given segmentation task, one can appropriately choose among the described elements to design an algorithm.

Most detection procedures described above are available within the Python language from `ruptures` (ctruong.perso.math.cnrs.fr/ruptures). The same framework is used to implement the algorithms.

Because of the approach adopted in this work, many methods were left off this review. The reader can find more information in the literature, for instance on Bayesian methods [48, 100], an in-depth theoretical survey [72] and numerical comparisons in controlled settings [44].

9. Algorithms

Algorithm 1 Algorithm Opt

Input: signal $\{y_t\}_{t=1}^{T}$, cost function $c(\cdot)$, number of regimes $K \geq 2$.
for all $(u, v),\ 1 \leq u < v \leq T$ **do**
  Initialize $C_1(u, v) \leftarrow c(\{y_t\}_{t=u}^{v})$.
end for
for $k = 2, \dots, K-1$ **do**
  for all $u, v \in \{1, \dots, T\}, v - u \geq k$ **do**
    $C_k(u, v) \leftarrow \min_{u+k-1 \leq t < v} C_{k-1}(u, t) + C_1(t+1, v)$
  end for
end for
Initialize L, a list with K elements.
Initialize the last element: $L[K] \leftarrow T$.
Initialize $k \leftarrow K$.
while $k > 1$ **do**
  $s \leftarrow L(k)$
  $t^\star \leftarrow \arg\min_{k-1 \leq t < s} C_{k-1}(1, t) + C_1(t+1, s)$
  $L(k-1) \leftarrow t^\star$
  $k \leftarrow k - 1$
end while
Remove T from L
Output: set L of estimated breakpoint indexes.

Name	Language	Link	Type(s) of change	Algorithm(s)
wbsts [83, 84]	R	cran.r-project.org/package=wbsts	Change in 2^{nd} order stationary structure [105]	BinSeg [52, 83]
trend [113]	R	cran.r-project.org/package=trend	Single shift in trend	Opt
strucchange [136]	R	cran.r-project.org/package=strucchange	Shifts in mean and in linear model	Opt
SeqCBS [124]	R	cran.r-project.org/package=SeqCBS	Mean-shifts in Poisson processes	BinSeg [106]
SegCorr [41]	R	cran.r-project.org/package=SegCorr	Shifts in mean and scale	Opt
cpm [119]	R	cran.r-project.org/package=cpm	Change in Gaussian, exponential, Bernoulli random variables, and general distribution change	Win, BinSeg
not [20]	R	cran.r-project.org/package=not	Mean-shifts in univariate signals (with different types of noise)	BinSeg
factorcpt [37]	R	cran.r-project.org/package=factorcpt	Mean-shifts with factor analysis	BinSeg
ecp [71]	R	cran.r-project.org/package=ecp	Distribution changes	BinSeg, BotUp
changepoint [80]	R	cran.r-project.org/package=changepoint	Mean and scale shifts in univariate signals	Pelt
breakfast [53]	R	cran.r-project.org/package=breakfast	Mean-shifts in univariate signals	BinSeg [50–52]
bcp [46]	R	cran.r-project.org/package=bcp	Bayesian counterpart of strucchange	Bayesian
changepoint.np [66]	R	cran.r-project.org/package=changepoint.np	Change in distribution (based on empirical distribution function)	Pelt
Segmentor3IsBack [39]	R	cran.r-project.org/package=Segmentor3IsBack	Distribution changes for Gaussian, Poisson, exponential, negative binomial variables	Opt
wbs [19]	R	cran.r-project.org/package=wbs	Mean-shifts in univariate signals	BinSeg [52]
AR1seg [30, 31]	R	cran.r-project.org/package=AR1seg	Mean-shifts in AR(1) processes	Opt

Table 3: Summary of available libraries for change point detection.

Algorithm 2 Algorithm BinSeg

Input: signal $\{y_t\}_{t=1}^T$, cost function $c(\cdot)$, stopping criterion.
Initialize $L \leftarrow \{\ \}$. ▷ Estimated breakpoints.
repeat
  $k \leftarrow |L|$. ▷ Number of breakpoints
  $t_0 \leftarrow 0$ and $t_{k+1} \leftarrow T$ ▷ Dummy variables.
  if $k > 0$ **then**
    Denote by t_i $(i = 1, \dots, k)$ the elements (in ascending order) of L, i.e. $L = \{t_1, \dots, t_k\}$.
  end if
  Initialize G a $(k+1)$-long array. ▷ list of gains
  for $i = 0, \dots, k$ **do**
    $G[i] \leftarrow c(y_{t_i..t_{i+1}}) - \min_{t_i < t < t_{i+1}} [c(y_{t_i..t}) + c(y_{t..t_{i+1}})]$.
  end for
  $\widehat{i} \leftarrow \arg\max_i G[i]$
  $\hat{t} \leftarrow \underset{t_{\widehat{i}} < t < t_{\widehat{i}+1}}{\arg\min}\ [c(y_{t_{\widehat{i}}..t}) + c(y_{t..t_{\widehat{i}+1}})]$.
  $L \leftarrow L \cup \{\hat{t}\}$
until stopping criterion is met.
Output: set L of estimated breakpoint indexes.

Algorithm 3 Algorithm BotUp

Input: signal $\{y_t\}_{t=1}^T$, cost function $c(\cdot)$, stopping criterion, grid size $\delta > 2$.
Initialize $L \leftarrow \{\delta, 2\delta, \dots, (\lfloor T/\delta \rfloor - 1)\,\delta\}$. ▷ Estimated breakpoints.
repeat
  $k \leftarrow |L|$. ▷ Number of breakpoints
  $t_0 \leftarrow 0$ and $t_{k+1} \leftarrow T$ ▷ Dummy variables.
  Denote by t_i $(i = 1, \dots, k)$ the elements (in ascending order) of L, i.e. $L = \{t_1, \dots, t_k\}$.
  Initialize G a $(k-1)$-long array. ▷ list of gains
  for $i = 1, \dots, k-1$ **do**
    $G[i-1] \leftarrow c(y_{t_{i-1}..t_{i+1}}) - [c(y_{t_{i-1}..t_i}) + c(y_{t_i..t_{i+1}})]$.
  end for
  $\widehat{i} \leftarrow \arg\min_i G[i]$
  Remove $t_{\widehat{i}+1}$ from L.
until stopping criterion is met.
Output: set L of estimated breakpoint indexes.

Algorithm 4 Algorithm Win

Input: signal $\{y_t\}_{t=1}^T$, cost function $c(\cdot)$, half-window width w, peak search procedure PKSearch.
Initialize $Z \leftarrow [0, 0, \dots]$ a T-long array filled with 0. ▷ Score list.
for $t = w, \dots, T - w$ **do**
    $p \leftarrow (t - w)..t$.
    $q \leftarrow t..(t + w)$.
    $r \leftarrow (t - w)..(t + w)$.
    $Z[t] \leftarrow c(y_r) - [c(y_p) + c(y_q)]$.
end for
$L \leftarrow \text{PKSearch}(Z)$ ▷ Peak search procedure.
Output: set L of estimated breakpoint indexes.

Algorithm 5 Algorithm Pelt

Input: signal $\{y_t\}_{t=1}^T$, cost function $c(\cdot)$, penalty value β.
Initialize Z a $(T + 1)$-long array; $Z[0] \leftarrow -\beta$.
Initialize $L[0] \leftarrow \emptyset$.
Initialize $\chi \leftarrow \{0\}$. ▷ Admissible indexes.
for $t = 1, \dots, T$ **do**
    $\hat{t} \leftarrow \arg\min_{s \in \chi} \big[Z[s] + c(y_{s..t}) + \beta\big]$.
    $Z[t] \leftarrow \big[Z[\hat{t}] + c(y_{\hat{t}..t}) + \beta\big]$
    $L[t] \leftarrow L[\hat{t}] \cup \{\hat{t}\}$.
    $\chi \leftarrow \{s \in \chi \ : \ Z[s] + c(y_{s..t}) \leq Z[t]\} \cup \{t\}$
end for
Output: set $L[T]$ of estimated breakpoint indexes.

References

[1] S. Adak. Time-dependent spectral analysis of nonstationary time series. *Journal of the American Statistical Association*, 93(444):1488–1501, 1998.

[2] D. Angelosante and G. B. Giannakis. Group lassoing change-points piece-constant AR processes. *EURASIP Journal on Advances in Signal Processing*, 70, 2012.

[3] S. Arlot and A. Celisse. A survey of cross-validation procedures for model selection. *Statistical Surveys*, 4:40–79, 2010.

[4] S. Arlot, A. Celisse, and Z. Harchaoui. Kernel change-point detection. *arXiv preprint arXiv:1202.3878*, pages 1–26, 2012.

[5] A. Aue and L. Horvàth. Structural breaks in time series. *Journal of Time Series Analysis*, 34:1–16, 2012.

[6] J. Bai. Least squares estimation of a shift in linear processes. *Journal of Time Series Analysis*, 15(5):453–472, 1994.

[7] J. Bai. Least absolute deviation of a shift. *Econometric Theory*, 11(3):403–436, 1995.

[8] J. Bai. Testing for parameter constancy in linear regressions: an empirical distribution function approach. *Econometrica*, 64(3):597–622, 1996.

[9] J. Bai. Estimating multiple breaks one at a time. *Econometric Theory*, 13(3):315–352, 1997.

[10] J. Bai. Estimation of a change-point in multiple regression models. *Review of Economic and Statistics*, 79(4): 551–563, 1997.

[11] J. Bai. Estimation of multiple-regime regressions with least absolutes deviation. *Journal of Statistical Planning and Inference*, 74:103–134, 1998.

[12] J. Bai. Likelihood ratio tests for multiple structural changes. *Journal of Econometrics*, 91(2):299–323, 1999.

[13] J. Bai. Vector autoregressive models with structural changes in regression coefficients and in variance–covariance matrices. *Annals of Economics and Finance*, 1(2):301–336, 2000.

[14] J. Bai. Common breaks in means and variances for panel data. *Journal of Econometrics*, 157:78–92, 2010.

[15] J. Bai and P. Perron. Estimating and testing linear models with multiple structural changes. *Econometrica*, 66(1): 47–78, 1998.

[16] J. Bai and P. Perron. Critical values for multiple structural change tests. *Econometrics Journal*, 6(1):72–78, 2003.

[17] J. Bai and P. Perron. Multiple structural change models: a simulation analysis. *Journal of Applied Econometrics*, 18:1–22, 2003.

[18] J. Bai and P. Perron. Computation and analysis of multiple structural change models. *Journal of applied econometrics*, 18(1):1–22, 2003.

[19] R. Baranowski and P. Fryzlewicz. wbs: wild binary segmentation for multiple change-point detection, 2015. URL https://cran.r-project.org/package=wbs.

[20] R. Baranowski, Y. Chen, and P. Fryzlewicz. not: narrowest-over-threshold change-point detection, 2016. URL https://cran.r-project.org/package=not.

[21] L. Bardwell and P. Fearnhead. Bayesian detection of abnormal segments in multiple time series. *Bayesian Analysis*, 12(1):193–218, 2017.

[22] M. Basseville and I. Nikiforov. *Detection of abrupt changes: theory and application*, volume 104. Prentice Hall Englewood Cliffs, 1993.

[23] R. Bellman. On a routing problem. *Quaterly of Applied Mathematics*, 16(1):87–90, 1955.

[24] L. Birgé and P. Massart. Gaussian model selection. *Journal of the European Mathematical Society*, 3(3):203–268, 2001.

[25] L. Birgé and P. Massart. Minimal penalties for Gaussian model selection. *Probability Theory and Related Fields*, 138(1):33–73, 2007.

[26] L. Boysen, A. Kempe, V. Liebscher, A. Munk, and O. Wittich. Consistencies and rates of convergence of jump-penalized least squares estimators. *The Annals of Statistics*, 37(1):157–183, 2009.

[27] B. E. Brodsky and B. S. Darkhovsky. *Nonparametric methods in change point problems*. Springer Netherlands, 1993.

[28] B. E. Brodsky, B. S. Darkhovsky, A. Y. Kaplan, and S. L. Shishkin. A nonparametric method for the segmentation of the EEG. *Computer Methods and Programs in Biomedicine*, 60(2):93–106, 1999.

[29] O. Cappé, É. Moulines, and T. Rydén. *Inference in hidden Markov models*. Springer-Verlag New York, 2005.

[30] S. Chakar, É. Lebarbier, C. Levy-Leduc, and S. Robin. AR1seg: segmentation of an autoregressive Gaussian process of order 1, 2014. URL https://cran.r-project.org/package=AR1seg.

[31] S. Chakar, É. Lebarbier, C. Levy-Leduc, and S. Robin. A robust approach for estimating change-points in the mean of an AR(1) process. *Bernouilli Society for Mathematical Statistics and Probability*, 23(2):1408–1447, 2017.

[32] J. Chen and A. K. Gupta. Testing and locating variance changepoints with application to stock prices. *Journal of the American Statistical Association*, 92(438):739–747, 1997.

[33] J. Chen and A. K. Gupta. *Parametric Statistical Change Point Analysis: With Applications to Genetics, Medicine, and Finance*. 2011.

[34] S. S. Chen and P. S. Gopalakrishnan. Speaker, environment and channel change detection and clustering via the bayesian information criterion. In *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, page 8, Landsdowne, VA, 1998.
[35] H Chernoff and S Zacks. Estimating the Current Mean of a Normal Distribution which is Subjected to Changes in Time. *The Annals of Mathematical Statistics*, 35(3):999–1018, 1964.
[36] S. Chib. Estimation and comparison of multiple change-point models. *Journal of Econometrics*, 86(2):221–241, 1998.
[37] H. Cho, M. Barigozzi, and P. Fryzlewicz. factorcpt: simultaneous change-point and factor analysis, 2016. URL https://cran.r-project.org/package=factorcpt.
[38] S. Clemencon, M. Depecker, and N. Vayatis. AUC optimization and the two-sample problem. In *Advances in Neural Information Processing Systems 22 (NIPS 2009)*, pages 360–368, Vancouver, Canada, 2009.
[39] A. Cleynen, G. Rigaill, and M. Koskas. Segmentor3IsBack: a fast segmentation algorithm, 2016. URL https://cran.r-project.org/package=Segmentor3IsBack.
[40] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th International Conference on Machine Learning (ICML)*, pages 209–216, Corvalis, Oregon, USA, 2007.
[41] E. I. Delatola, É. Lebarbier, T. Mary-Huard, F. Radvanyi, S. Robin, and J. Wong. SegCorr: a statistical procedure for the detection of genomic regions of correlated expression. *BMC Bioinformatics*, 18(1):1–15, 2017.
[42] F. Desobry, M. Davy, and C. Doncarli. An online kernel change detection algorithm. *IEEE Transactions on Signal Processing*, 53(8):2961–2974, 2005.
[43] B. M. Doyle and J. Faust. Breaks in the variability and comovement of G-7 economic growth. *The Review of Economics and Statistics*, 87(4):721–740, 2005.
[44] I. A. Eckley, P. Fearnhead, and R. Killick. Analysis of changepoint models. In *Bayesian Time Series Models*, pages 205–224. Cambridge University Press, 2011.
[45] B. Efron, T. Hastie, and R. Tibshirani. Least angle regression. *The Annals of Statistics*, 32(1):407–499, 2004.
[46] C. Erdman and J. W. Emerson. bcp: an R package for performing a Bayesian analysis of change point problems. *Journal of Statistical Software*, 23(3):1–13, 2007.
[47] R. Esteller, G. Vachtsevanos, J. Echauz, and B. Litt. A Comparison of waveform fractal dimension algorithms. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 48(2):177–183, 2001.
[48] P Fearnhead. Exact and efficient Bayesian inference for multiple changepoint problems. *Statistics and Computing*, 16(2):203–213, 2006.
[49] P. Fragkou, V. Petridis, and Ath Kehagias. A dynamic programming algorithm for linear text segmentation. *Journal of Intelligent Information Systems*, 23(2):179–197, 2004.
[50] K. Frick, A. Munk, and H. Sieling. Multiscale change point inference. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 76(3):495–580, 2014.
[51] P. Fryzlewicz. Unbalanced Haar technique for nonparametric function estimation. *Journal of the American Statistical Association*, 102(480):1318–1327, 2007.
[52] P. Fryzlewicz. Wild binary segmentation for multiple change-point detection. *The Annals of Statistics*, 42(6): 2243–2281, 2014. doi: 10.1214/14-AOS1245.
[53] P. Fryzlewicz. breakfast: multiple change-point detection and segmentation, 2017. URL https://cran.r-project.org/package=breakfast.
[54] P. Fryzlewicz and S. Subba Rao. Multiple change-point detection for auto-regressive conditional heteroscedastic processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(5):903–924, 2014.
[55] D. Garreau and S. Arlot. Consistent change-point detection with kernels. *arXiv preprint arXiv:1612.04740*, pages 1–41, 2016.
[56] A. Gretton, K. Borgwardt, M. J. Rasch, B. Scholkopf, and A. J. Smola. A kernel method for the two-sample problem. *arXiv preprint arXiv:0805.2368*, 2008.
[57] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola. A kernel two-sample test. *Journal of Machine Learning Research (JMLR)*, 13:723–773, 2012.
[58] Y. Guédon. Exploring the latent segmentation space for the assessment of multiple change-point models. *Computational Statistics*, 28(6):2641–2678, 2013.
[59] Z. Harchaoui and O. Cappé. Retrospective mutiple change-point estimation with kernels. In *Proceedings of the IEEE/SP Workshop on Statistical Signal Processing*, pages 768–772, Madison, Wisconsin, USA, 2007.
[60] Z. Harchaoui and C. Lévy-Leduc. Multiple Change-Point Estimation With a Total Variation Penalty. *Journal of the American Statistical Association*, 105(492):1480–1493, 2010.
[61] Z. Harchaoui and C. Lévy-Leduc. Multiple change-point estimation with a total variation penalty. *Journal of the American Statistical Association*, 105(492):1480–1493, 2010.
[62] Z. Harchaoui, F. Bach, and É. Moulines. Kernel change-point analysis. In *Advances in Neural Information Processing Systems 21 (NIPS 2008)*, pages 609–616, Vancouver, Canada, 2008.
[63] Z. Harchaoui, F. Vallet, A. Lung-Yut-Fong, and O. Cappé. A regularized kernel-based approach to unsupervised

audio segmentation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1665–1668, Taipei, Taiwan, 2009.

[64] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*, volume 1. 2009.

[65] K. Haynes, I. A. Eckley, and P. Fearnhead. Computationally efficient changepoint detection for a range of penalties. *Journal of Computational and Graphical Statistics*, 26(1):134–143, 2017.

[66] K. Haynes, P. Fearnhead, and I. A. Eckley. A computationally efficient nonparametric approach for changepoint detection. *Statistics and Computing*, 27:1293–1305, 2017.

[67] J. Himberg, K. Korpiaho, H. Mannila, J. Tikanmaki, and H. T. Toivonen. Time series segmentation for context recognition in mobile devices. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, pages 203–210, 2001.

[68] D. V. Hinkley. Inference about the change point in a sequence of random variables. *Biometrika*, 57(1):1–17, 1970.

[69] T. Hocking, G Rigaill, J.-P. Vert, and F. Bach. Learning sparse penalties for change-point detection using max margin interval regression. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 172–180, Atlanta, USA, 2013.

[70] T. Hocking, G. Schleiermacher, I. Janoueix-Lerosey, V. Boeva, J. Cappo, O. Delattre, F. Bach, and J.-P. Vert. Learning smoothing models of copy number profiles using breakpoint annotations. *BMC Bioinformatics*, 14(1):164, 2013.

[71] N. A. James and D. S. Matteson. ecp: an R package for nonparametric multiple change point analysis of multivariate data. *Journal of Statistical Software*, 62(7):1–25, 2014.

[72] V. Jandhyala, S. Fotopoulos, I. Macneill, and P. Liu. Inference for single and multiple change-points in time series. *Journal of Time Series Analysis*, 34(4):423–446, 2013.

[73] K. Karagiannaki, A. Panousopoulou, and P. Tsakalides. An online feature selection architecture for Human Activity Recognition. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2522–2526, New Orleans, LA, USA, 2017.

[74] Y. Kawahara. Change-point detection in time-series data by direct density-ratio estimation. *Proceedings of the 2009 SIAM International Conference on Data Mining (ICDM)*, pages 389–400, 2009.

[75] S M Kay. *Fundamentals of statistical signal processing: Detection theory*. Prentice Hall, 1998.

[76] S. M. Kay and A. V. Oppenheim. *Fundamentals of Statistical Signal Processing, Volume II: Detection Theory*. Prentice Hall, 1993.

[77] E. Keogh, S. Chu, D. Hart, and M. Pazzani. An online algorithm for segmenting time series. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, pages 289–296, 2001.

[78] E. Keogh, S. Chu, D. Hart, and M. Pazzani. Segmenting time series: a survey and novel approach. *Data Mining in Time Series Databases*, 57(1):1–22, 2004.

[79] D. Kifer, S. Ben-David, and J. Gehrke. Detecting change in data streams. In *Proceedings of the Thirtieth International Conference on Very Large Data Bases (VLDB) - Volume 30*, pages 180–191, Toronto, Canada, 2004.

[80] R. Killick and I. A. Eckley. changepoint: an R package for changepoint analysis. *Journal of Statistical Software*, 58(3):1–19, 2014.

[81] R. Killick, P. Fearnhead, and I. Eckley. Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, 107(500):1590–1598, 2012.

[82] S. I. M. Ko, T. T. L. Chong, and P. Ghosh. Dirichlet process hidden Markov multiple change-point model. *Bayesian Analysis*, 10(2):275–296, 2015.

[83] K. Korkas and P. Fryzlewicz. wbsts: multiple change-point detection for nonstationary time series, 2015. URL https://cran.r-project.org/package=wbsts.

[84] K. Korkas and P. Fryzlewicz. Multiple change-point detection for non-stationary time series using wild binary segmentation. *Statistica Sinica*, 27(1):287–311, 2017.

[85] P. R. Krishnaiah. Review about estimation of change points. *Handbook of Statistics*, 7:375–402, 1988.

[86] W. R. Lai, M. D. Johnson, R. Kucherlapati, and P. J. Park. Comparative analysis of algorithms for identifying amplifications and deletions in array CGh data. *Bioinformatics*, 21(19):3763–3770, 2005.

[87] R. Lajugie, F. Bach, and S. Arlot. Large-margin mMetric learning for constrained partitioning problems. In *Proceedings of the 31st International Conference on Machine Learning (ICML)*, pages 297–395, Beijing, China, 2014.

[88] M. Lavielle. Optimal segmentation of random processes. *IEEE Transactions on Signal Processing*, 46(5):1365–1373, 1998.

[89] M. Lavielle. Detection of multiples changes in a sequence of dependant variables. *Stochastic Processes and their Applications*, 83(1):79–102, 1999.

[90] M. Lavielle. Using penalized contrasts for the change-point problem. *Signal Processing*, 85(8):1501–1510, 2005.

[91] M. Lavielle and É. Moulines. Least-squares estimation of an unknown number of shifts in a time series. *Journal of Time Series Analysis*, 21(1):33–59, 2000.

[92] M. Lavielle and G. Teyssière. Adaptive detection of multiple change-points in asset price volatility. In *Long-Memory in Economics*, pages 129–156. Springer Verlag, Berlin, Germany, 2007.

[93] É. Lebarbier. Detecting multiple change-points in the mean of gaussian process by model selection. *Signal Processing*, 85(4):717–736, 2005.
[94] C. Lévy-Leduc and F. Roueff. Detection and localization of change-points in high-dimensional network traffic data. *The Annals of Applied Statistics*, 3(2):637–662, 2009.
[95] Song Liu, Makoto Yamada, Nigel Collier, and Masashi Sugiyama. Change-point detection in time-series data by relative density-ratio estimation. *Neural Networks*, 43:72–83, 2013.
[96] G. Lorden. Procedures for reacting to a change in distribution. *The Annals of Mathematical Statistics*, 42(6): 1897–1908, 1971.
[97] A. Lung-Yut-Fong, C. Lévy-Leduc, and O. Cappé. Distributed detection/localization of change-points in high-dimensional network traffic data. *Statistics and Computing*, 22(2):485–496, 2012.
[98] P. C. Mahalanobis. On the generalised distance in statistics. *Proceedings of the National Institute of Sciences of India*, 2(1):49–55, 1936.
[99] R. Maidstone. Efficient Analysis of Complex Changepoint Models. page 34, 2013.
[100] R. Maidstone, T. Hocking, G. Rigaill, and P. Fearnhead. On optimal multiple changepoint algorithms for large data. *Statistics and Computing*, 27(2):519–533, 2017.
[101] C. L. Mallows. Some comments on Cp. *Technometrics*, 15(4):661–675, 1973.
[102] I. Melnyk and A. Banerjee. A spectral algorithm for inference in hidden semi-Markov models. *Journal of Machine Learning Research (JMLR)*, 18:1–39, 2017.
[103] É. Moulines, F. Bach, and Z. Harchaoui. Testing for homogeneity with kernel fisher discriminant analysis. In *Advances in Neural Information Processing Systems 20 (NIPS 2008)*, pages 609–616, 2008.
[104] C. F. H. Nam, J. A. D. Aston, and A. M. Johansen. Quantifying the uncertainty in change points. *Journal of Time Series Analysis*, 33:807–823, 2012.
[105] G. P. Nason, R. von Sachs, and G. Kroisandt. Wavelet processes and adaptive estimation of the evolutionary wavelet spectrum. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 62(2):271–292, 2000.
[106] A. B. Olshen, E. S. Venkatraman, R. Lucito, and M. Wigler. Circular binary segmentation for the analysis of array-based DNA copy number data. *Biostatistics*, 5(4):557–572, 2004.
[107] L. Oudre, A. Lung-Yut-Fong, and P. Bianchi. Segmentation of accelerometer signals recorded during continuous treadmill walking. In *Proceedings of the 19th European Signal Processing Conference (EUSIPCO)*, pages 1564–1568, 2011.
[108] E. S. Page. A test for a change in a parameter occurring at an unknown point. *Biometrika*, 42:523–527, 1955.
[109] V. Pavlovic, J. M. Rehg, and J. MacCormick. Learning Switching Linear Models of Human Motion. In *Advances in Neural Information Processing Systems 13 (NIPS 2000)*, pages 981–987, Denver, United States, 2000.
[110] B. S. Paye and A. Timmermann. Instability of return prediction models. *Journal of Empirical Finance*, 13(3): 274–315, 2006.
[111] P. Perron. Dealing with structural breaks. *Palgrave handbook of econometrics*, 1(2):278–352, 2006.
[112] F. Picard, S. Robin, M. Lavielle, C. Vaisse, and J.-J. Daudin. A statistical approach for array CGH data analysis. *BMC Bioinformatics*, 6(1):27, 2005.
[113] T. Pohlert. trend: non-parametric trend tests and change-point detection, 2017. URL https://cran.r-project.org/package=trend.
[114] R. Prescott Adams and D. J. C. MacKay. Bayesian Online Changepoint Detection. Technical report, 2007.
[115] Z. Qu and P. Perron. Estimating and testing structural changes in multivariate regressions. *Econometrica*, 75(2): 459–502, 2007.
[116] L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.
[117] J. Reeves, J. Chen, X. L. Wang, R. Lund, and Q. Q. Lu. A review and comparison of changepoint detection techniques for climate data. *Journal of Applied Meteorology and Climatology*, 46(6):900–915, 2007. ISSN 15588424. doi: 10.1175/JAM2493.1.
[118] G. Rigaill. A pruned dynamic programming algorithm to recover the best segmentations with 1 to K_max change-points. *Journal de la Société Française de Statistique*, 156(4):180–205, 2015.
[119] G. Ross. Parametric and nonparametric sequential change detection in R: the cpm package. *Journal of Statistical Software*, 66(3):1–20, 2015.
[120] B. Schölkopf and A. J. Smola. *Learning with kernels*. MIT Press, Cambridge, USA, 2002.
[121] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978.
[122] N. Seichepine, S. Essid, C. Fevotte, and O. Cappé. Piecewise constant nonnegative matrix factorization. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6721–6725, Florence, Italy, 2014.
[123] A. Sen and M. S. Srivastava. On tests for detecting change in mean. *The Annals of Statistics*, 3(1):98–108, 1975.
[124] J. J. Shen and N. R. Zhang. SeqCBS: CN profiling using sequencing and CBS, 2012.
[125] David Siegmund and E S Venkatraman. Using the generalized likelihood ratio statistic for sequential detection

of a change-point. *The Annals of Statistics*, pages 255–271, 1995.

[126] B. K. Sriperumbudur, A. Gretton, K. Fukumizu, G. Lanckriet, and B. Schölkopf. Injective Hilbert space embeddings of probability measures. In *Proceedings of the 21st Conference on Learning Theory (COLT)*, pages 9–12, Helsinki, Finland, 2008.

[127] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996.

[128] E. S. Venkatraman and A. B. Olshen. A faster circular binary segmentation algorithm for the analysis of array CGH data. *Bioinformatics*, 23(6):657–663, 2007.

[129] J.-P. Vert and K. Bleakley. Fast detection of multiple change-points shared by many signals using group LARS. In *Advances in Neural Information Processing Systems 23 (NIPS 2010)*, volume 1, pages 2343–2351, Vancouver, Canada, 2010.

[130] L. Y. Vostrikova. Detecting disorder in multidimensional random processes. *Soviet Math. Dokl.*, 24:55–59, 1981.

[131] H. Vullings, M. Verhaegen, and H. Verbruggen. ECG segmentation using time-warping. In *Lecture notes in computer science*, pages 275–286. Springer, 1997.

[132] H. Willenbrock and J. Fridlyand. A comparison study: applying segmentation to array CGH data for downstream analyses. *Bioinformatics*, 21(22):4084–4091, 2005.

[133] E. P. Xing, M. I. Jordan, and S. J. Russell. Distance metric learning, with application to clustering with side-Information. In *Advances in Neural Information Processing Systems 21 (NIPS 2003)*, pages 521–528, 2003.

[134] Y.-C. Yao. Estimating the number of change-points via Schwarz' criterion. *Statistics and Probability Letters*, 6(3): 181–189, 1988.

[135] Y.-C. Yao and S. T. Au. Least-squares estimation of a step function. *Sankhyā: The Indian Journal of Statistics, Series A*, 51(3):370–381, 1989.

[136] A. Zeileis, F. Leisch, K. Hornik, and C. Kleiber. strucchange: an R package for testing for structural change in linear regression models. *Journal of Statistical Software*, 7(2):1–38, 2002.

[137] N. R. Zhang and D. O. Siegmund. A modified Bayes information criterion with applications to the analysis of comparative genomic hybridization data. *Biometrics*, 63(1):22–32, 2007.