

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/335937522>

Selective review of offline change point detection methods

Article in *Signal Processing* · September 2019

DOI: 10.1016/j.sigpro.2019.107299

CITATIONS

899

READS

2,157

3 authors:



Charles Truong

22 PUBLICATIONS 1,013 CITATIONS

SEE PROFILE



Laurent Oudre

Ecole Normale Supérieure Paris-Saclay

103 PUBLICATIONS 1,790 CITATIONS

SEE PROFILE



Nicolas Vayatis

Ecole normale supérieure Paris-Saclay

212 PUBLICATIONS 4,194 CITATIONS

SEE PROFILE

Selective review of offline change point detection methods

Charles Truong^{a,*}, Laurent Oudre^b, Nicolas Vayatis^a

^a*CMLA, CNRS, ENS Paris Saclay*

^b*L2TI, University Paris 13*

Abstract

This article presents a selective survey of algorithms for the offline detection of multiple change points in multivariate time series. A general yet structuring methodological strategy is adopted to organize this vast body of work. More precisely, detection algorithms considered in this review are characterized by three elements: a cost function, a search method and a constraint on the number of changes. Each of those elements is described, reviewed and discussed separately. Implementations of the main algorithms described in this article are provided within a Python package called ruptures.

Keywords: change point detection, segmentation, statistical signal processing

1. Introduction

A common task in signal processing is the identification and analysis of complex systems whose underlying state changes, possibly several times. This setting arises when industrial systems, physical phenomena or human activity are continuously monitored with sensors. The objective of practitioners is to extract from the recorded signals a posteriori meaningful information about the different states and transitions of the monitored object for analysis purposes. This setting encompasses a broad range of real-world scenarios and a wide variety of signals.

Change point detection is the task of finding changes in the underlying model of a signal or time series. The first works on change point detection go back to the 50s [1, 2]: the goal was to locate a shift in the mean of independent and identically distributed (iid) Gaussian variables for industrial quality control purposes. Since then, this problem has been actively investigated, and is periodically the subject of in-depth monographs [3–6]. This subject has generated important activity in statistics and signal processing [7–9] but also in various application settings such as speech processing [10–13], financial analysis [7, 14, 15], bio-informatics [16–24], climatology [25–27], network traffic data

*Corresponding author

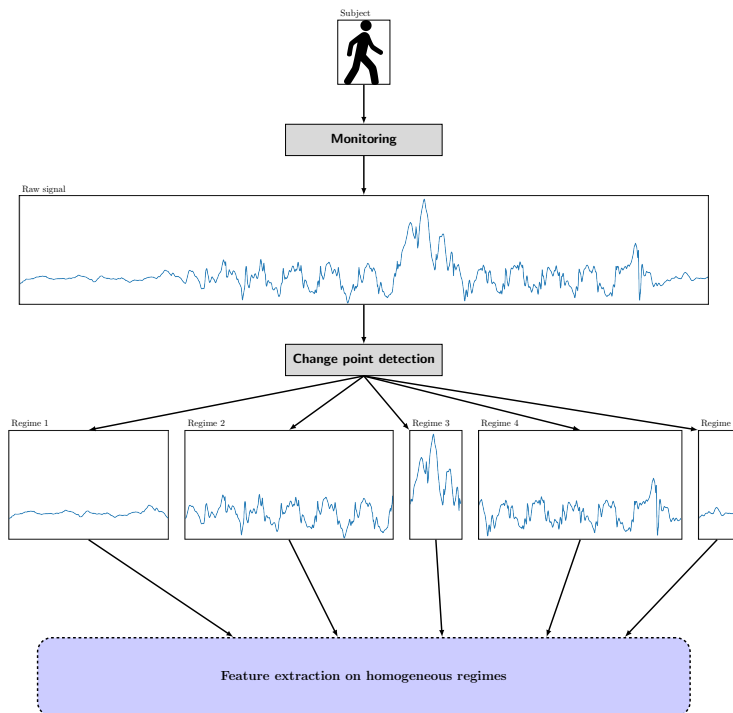


Figure 1: Flowchart of a study scheme, for gait analysis.

analysis [28, 29]. Modern applications in bioinformatics, finance, monitoring of complex systems have also motivated recent developments from the machine learning community [18, 30, 31].

Let us take the example of gait analysis, illustrated on the flowchart displayed on Figure 1. In this context, a patient’s movements are monitored with accelerometers and gyroscopes while performing simple activities, for instance walking at preferred speed, running or standing still. The objective is to objectively quantify gait characteristics [32–36]. The resulting signal is described as a succession of non-overlapping segments, each one corresponding to an activity and having its own gait characteristics. Insightful features from homogeneous phases can be extracted if the temporal boundaries of those segments are identified. This analysis therefore needs a preliminary processing of the signals: change point detection.

Change point detection methods are divided into two main branches: *online* methods, that aim to detect changes as soon as they occur in a real-time setting, and *offline* methods that retrospectively detect changes when all samples are received. The former task is often referred to as *event or anomaly detection*, while the latter is sometimes called *signal segmentation*.

In this article, we propose a survey of algorithms for the detection of multiple change points in multivariate time series. All reviewed methods presented in

39 this paper address the problem of *offline* (also referred to as *retrospective* or *a*
40 *posteriori*) change point detection, in which segmentation is performed after the
41 signal has been collected. The objective of this article is to facilitate the search
42 of a suitable detection method for a given application. In particular, focus is
43 made on practical considerations such as implementations and procedures to
44 calibrate the algorithms. This review also presents the mathematical properties
45 of the main approaches, as well as the metrics to evaluate and compare their
46 results. This article is linked with a Python scientific library called **ruptures**
47 [37], that includes a modular and easy-to-use implementation of a subset of the
48 methods presented in this paper.

49 2. Background

50 This section introduces the main concepts for change point detection, as well
51 as the selection criteria and the outline of this review.

52 2.1. Notations

53 In the remainder of this article, we use the following notations. For a given
54 signal $y = \{y_t\}_{t=1}^T$, the $(b-a)$ -sample long sub-signal $\{y_t\}_{t=a+1}^b$ ($1 \leq a < b \leq T$)
55 is simply denoted $y_{a..b}$; the complete signal is therefore $y = y_{0..T}$. A set of
56 indexes is denoted by a calligraphic letter: $\mathcal{T} = \{t_1, t_2, \dots\} \subset \{1, \dots, T\}$, and
57 its cardinal is $|\mathcal{T}|$. For a set of indexes $\mathcal{T} = \{t_1, \dots, t_K\}$, the dummy indexes
58 $t_0 := 0$ and $t_{K+1} := T$ are implicitly available.

59 2.2. Problem formulation

60 Let us consider a multivariate non-stationary random process $y = \{y_1, \dots, y_T\}$
61 that takes value in \mathbb{R}^d ($d \geq 1$) and has T samples. The signal y is assumed to be
62 piecewise stationary, meaning that some characteristics of the process change
63 abruptly at some unknown instants $t_1^* < t_2^* < \dots < t_{K^*}^*$. Change point detection
64 consists in estimating the indexes t_k^* . Depending on the context, the number
65 K^* of changes may or may not be known, in which case it has to be estimated
66 too.

67 Formally, change point detection is cast as a model selection problem, which
68 consists in choosing the best possible segmentation \mathcal{T} according to a quantitative
69 criterion $V(\mathcal{T}, y)$ that must be minimized. (The function $V(\mathcal{T}, y)$ is simply
70 denoted $V(\mathcal{T})$ when it is obvious from the context that it refers to the signal
71 y .) The choice of the criterion function $V(\cdot)$ depends on preliminary knowledge
72 on the task at hand.

73 In this work, we make the assumption that the criterion function $V(\mathcal{T})$ for
74 a particular segmentation is a sum of costs of all the segments that define the
75 segmentation:

$$V(\mathcal{T}, y) := \sum_{k=0}^K c(y_{t_k..t_{k+1}}) \quad (1)$$

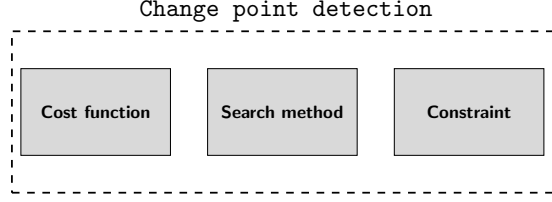


Figure 2: Typology of change point detection methods described in this article. Reviewed algorithms are defined by three elements: a cost function, a search method and a constraint (on the number of change points).

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 “best segmentation” $\hat{\mathcal{T}}$ is the minimizer of the criterion $V(\mathcal{T})$. In practice, depending on whether the number K^* of change points is known beforehand, change point detection methods fall into two categories.

- **Problem 1 : known number of changes K .** The change point detection problem with a fixed number K of change points consists in solving the following discrete optimization problem

$$\min_{|\mathcal{T}|=K} V(\mathcal{T}). \quad (\text{P1})$$

- **Problem 2 : unknown number of changes.** The change point detection problem with an unknown number of change points consists in solving the following discrete optimization problem

$$\min_{\mathcal{T}} V(\mathcal{T}) + \text{pen}(\mathcal{T}) \quad (\text{P2})$$

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

All change point detection methods considered in this work yield an exact or an approximate solution to either Problem 1 (P1) or Problem 2 (P2), with the function $V(\mathcal{T}, y)$ adhering to the format (1).

2.3. Selection criteria for the review

To better understand the strengths and weaknesses of change point detection methods, we propose to classify algorithms according to a comprehensive typology. Precisely, detection methods are expressed as the combination of the following three elements.

- **Cost function.** The cost function $c(\cdot)$ is a measure of “homogeneity”. Its choice encodes the type of changes that can be detected. Intuitively, $c(y_{a..b})$ is expected to be low if the sub-signal $y_{a..b}$ is “homogeneous” (meaning that it does not contain any change point), and large if the

sub-signal $y_{a..b}$ is “heterogeneous” (meaning that it contains one or several change points).

- **Search method.** The search method is the resolution procedure for the discrete optimization problems associated with Problem 1 (P1) and Problem 2 (P2). The literature contains several methods to efficiently solve those problems, in an exact fashion or in an approximate fashion. Each method strikes a balance between computational complexity and accuracy.
- **Constraint (on the number of change points).** When the number of changes is unknown (P2), a constraint is added, in the form of a complexity penalty $\text{pen}(\cdot)$ (P2), to balance out the goodness-of-fit term $V(\mathcal{T}, y)$. The choice of the complexity penalty is related to the amplitude of the changes to detect: with too “small” a penalty (compared to the goodness-of-fit) in (P2), many change points are detected, even those that are the result of noise. Conversely, too much penalization only detects the most significant changes, or even none.

This typology of change point detection methods is schematically shown on Figure 2.

2.4. Limitations

The described framework, however general, does not encompass all published change point detection methods. In particular, Bayesian approaches are not considered in the remainder of this article, even though they provide state-of-the-art results in several domains, such as speech and sound processing. The most well-known Bayesian algorithm is the Hidden Markov Model (HMM) [38]. This model was later extended, for instance with Dirichlet processes [39, 40] or product partition models [41, 42]. The interested reader can find reviews of Bayesian approaches in [4] and [6].

Also, several literature reviews with different selection criteria can be found. Recent and important works include [43] which focuses on window-based detection algorithms. In particular, the authors use the quantity of samples needed to detect a change as a basis for comparison. Maximum likelihood and Bayes-type detection are reviewed, from a theoretical standpoint, in [8]. Existing asymptotic distributions for change point estimates are described for several statistical models. In [44], detection is formulated as a statistical hypothesis testing problem, and emphasis is put on the algorithmic and theoretical properties of several sequential mean-shift detection procedures.

Finally, note that some of the cost functions presented in the present article only deal with univariate signals (see Section 4.1.2).

2.5. Outline of the article

Before starting this review, we propose in Section 3 a detailed overview of the main mathematical tools that can be used for evaluating and comparing the change point detection methods. The organization of the remaining of this

review article reflects the typology of change point detection methods, which is schematically shown on Figure 2. Precisely, the three defining elements of a detection algorithm are reviewed separately. In Section 4, cost functions from the literature are presented, along with the associated signal model and the type of change that can be detected. Whenever possible, theoretical results on asymptotic consistency are also given. Section 5 lists search methods that efficiently solve the discrete optimizations associated with Problem 1 (P1) and Problem 2 (P2). Both exact and approximate methods are described. Constraints on the number of change points are reviewed in Section 6. A summary table of the literature review can be found in Section 8. Additional considerations to apply change point detection in real-world scenarios are given in Section 7. The last section 9 is dedicated to the presentation of the Python package that goes with this article and propose a modular implementation of a subset of the approaches described in this article.

3. Evaluation

Change point detection methods can be evaluated either by proving some mathematical properties of the algorithms (such as consistency) in general case, or empirically by computing several metrics to assess the performances on a given dataset.

3.1. Consistency

A natural question when designing detection algorithms is the consistency of estimated change point indexes, as the number of samples T goes to infinity. In the literature, the “asymptotic setting” is intuitively described as follows: the observed signal y is regarded as a realization of a continuous-time process on an equispaced grid of size $1/T$, and “ T goes to infinity” means that the spacing of the sampling grid converges to 0. Precisely, for all $\tau \in [0, 1]$, let $Y(\tau)$ denote an \mathbb{R}^d -valued random variable such that

$$y_t = Y(t/T) \quad \forall t = 1, \dots, T. \quad (2)$$

The continuous-time process undergoes K^* changes in the probability distribution at the time instants $\tau_k^* \in (0, 1)$. Those τ_k^* are related to the change point indexes t_k^* through the following relationship:

$$t_k^* = \lfloor T\tau_k^* \rfloor. \quad (3)$$

Generally, for a given change point index t_k , the associated quantity $\tau_k = t_k/T \in (0, 1)$ is referred to as a change point *fraction*. In particular, the change point fractions τ_k^* ($k = 1, \dots, K^*$) of the time-continuous process Y are change point indexes of the discrete-time signal y . Note that in this asymptotic setting, the lengths of each regime of y increase linearly with T . The notion of asymptotic consistency of a change point detection method is formally introduced as follows.

178 **Definition 1 (Asymptotic consistency).** *A change point detection algorithm*
 179 *is said to be asymptotically consistent if the estimated segmentation $\widehat{\mathcal{T}} = \{\hat{t}_1, \hat{t}_2, \dots\}$*
 180 *satisfies the following conditions, when $T \rightarrow +\infty$:*

- 181 (i) $P(|\widehat{\mathcal{T}}| = K^*) \rightarrow 1,$
 182 (ii) $\frac{1}{T} \max \left\{ \max_{\hat{t} \in \widehat{\mathcal{T}}} \min_{t^* \in \mathcal{T}^*} |\hat{t} - t^*|, \max_{t^* \in \mathcal{T}^*} \min_{\hat{t} \in \widehat{\mathcal{T}}} |\hat{t} - t^*| \right\} \xrightarrow{p} 0,$

183 In Definition 1, the first condition is trivially verified when the number K^* of
 184 change points is known beforehand.

185 As for the second condition, it means that the estimated change point set
 186 $\widehat{\mathcal{T}}$ converges to \mathcal{T}^* , for a certain measure, which is often referred to as the
 187 Hausdorff distance, and is formally introduced later in the article.

188 More precisely, it is the estimated change point fractions that are consistent,
 189 and not the indexes themselves. In general, distances $|\hat{t} - t^*|$ between true change
 190 point indexes and their estimated counterparts do not converge to 0, even for
 191 simple models [18, 45–47]. As a result, consistency results in the literature only
 192 deal with change point fractions.

193 3.2. Evaluation metrics

194 Several metrics from the literature are presented below. Each metric cor-
 195 respond to one of the previously listed criteria by which segmentation perfor-
 196 mances are assessed. In the following, the set of true change points is denoted
 197 by $\mathcal{T}^* = \{t_1^*, \dots, t_{K^*}^*\}$, and the set of estimated change points is denoted by
 198 $\widehat{\mathcal{T}} = \{\hat{t}_1, \dots, \hat{t}_{\widehat{K}}\}$. Note that the cardinals of each set, K^* and \widehat{K} , are not
 199 necessarily equal.

200 3.2.1. Annotation error

201 The annotation error, denoted Δ_{AE} , is simply the difference between the
 202 predicted number of change points $|\widehat{\mathcal{T}}| (= \widehat{K})$ and the true number of change
 203 points $|\mathcal{T}^*| (= K^*)$:

$$\Delta_{\text{AE}}(\mathcal{T}^*, \widehat{\mathcal{T}}) := |\widehat{K} - K^*|. \quad (4)$$

204 This metric can be used to discriminate detection methods when the number
 205 of changes is unknown.

206 3.2.2. Hausdorff

207 The Hausdorff metric, denoted Δ_{HA} , measures the robustness of detection
 208 methods [47, 48]. Formally, it is equal to the greatest temporal distance between
 209 a change point and its prediction:

$$\Delta_{\text{HA}}(\mathcal{T}^*, \widehat{\mathcal{T}}) := \max \left\{ \max_{\hat{t} \in \widehat{\mathcal{T}}} \min_{t^* \in \mathcal{T}^*} |\hat{t} - t^*|, \max_{t^* \in \mathcal{T}^*} \min_{\hat{t} \in \widehat{\mathcal{T}}} |\hat{t} - t^*| \right\}. \quad (5)$$

210 It is the worst error made by the algorithm that produced $\widehat{\mathcal{T}}$ and is expressed
 211 in number of samples. If this metric is equal to zero, both breakpoint sets are

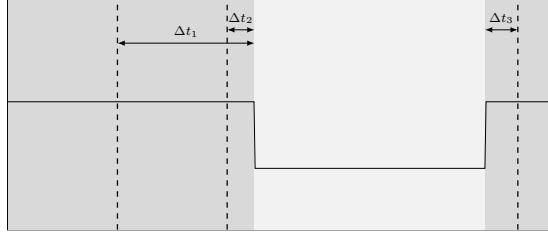


Figure 3: Hausdorff metric: Δ_{HA} . Alternating gray areas mark the segmentation \mathcal{T}^* ; dashed lines mark the segmentation $\hat{\mathcal{T}}$. Here, Δ_{HA} is equal to $\Delta t_1 = \max(\Delta t_1, \Delta t_2, \Delta t_3)$.

212 equal; it is large when a change point from either \mathcal{T}^* or $\hat{\mathcal{T}}$ is far from every
 213 change point of $\hat{\mathcal{T}}$ or \mathcal{T}^* respectively. Over-segmentation as well as under-
 214 segmentation is penalized. An illustrative example is displayed on Figure 3.
 215

216 3.2.3. Rand index

Accuracy can be measured by the Rand index, denoted Δ_{RI} , which is the average similarity between the predicted breakpoint set $\hat{\mathcal{T}}$ and the ground truth \mathcal{T}^* [30]. Intuitively, it is equal to the number of agreements between two segmentations. An agreement is a pair of indexes which are either in the same segment according to both $\hat{\mathcal{T}}$ and \mathcal{T}^* or in different segments according to both $\hat{\mathcal{T}}$ and \mathcal{T}^* . Formally, for a breakpoint set \mathcal{T} , the set of grouped indexes and the set of non-grouped indexes are respectively $\text{gr}(\mathcal{T})$ and $\text{ngr}(\mathcal{T})$:

$$\begin{aligned} \text{gr}(\mathcal{T}) &:= \{(s, t), 1 \leq s < t \leq T \text{ s.t. } s \text{ and } t \text{ belong to} \\ &\quad \text{the same segment according to } \mathcal{T} \}, \\ \text{ngr}(\mathcal{T}) &:= \{(s, t), 1 \leq s < t \leq T \text{ s.t. } s \text{ and } t \text{ belong to} \\ &\quad \text{different segments according to } \mathcal{T} \}. \end{aligned}$$

217 The Rand index is then defined as follows:

$$\Delta_{\text{RI}}(\mathcal{T}^*, \hat{\mathcal{T}}) := \frac{|\text{gr}(\hat{\mathcal{T}}) \cap \text{gr}(\mathcal{T}^*)| + |\text{ngr}(\hat{\mathcal{T}}) \cap \text{ngr}(\mathcal{T}^*)|}{T(T-1)}. \quad (6)$$

218 It is normalized between 0 (total disagreement) and 1 (total agreement).
 219 Originally, the Rand index has been introduced to evaluate clustering meth-
 220 ods [30, 47]. An illustrative example is displayed on Figure 4.

221 3.2.4. F1-score

222 Another measure of accuracy is the F1-score, denoted Δ_{F1} . Precision is the
 223 proportion of predicted change points that are true change points. Recall is
 224 the proportion of true change points that are well predicted. A breakpoint is
 225 considered detected up to a user-defined margin of error $M > 0$; true positives
 226 TP are true change points for which there is an estimated one at less than M

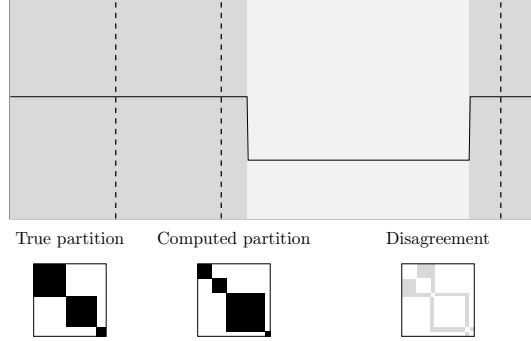


Figure 4: Rand index: Δ_{RI} . Top: alternating gray areas mark the segmentation \mathcal{T}^* ; dashed lines mark the segmentation $\hat{\mathcal{T}}$. Below: representations of associated adjacency matrices and disagreement matrix. The adjacency matrix of a segmentation is the $T \times T$ binary matrix with coefficient (s, t) equal to 1 if s and t belong to the same segment, 0 otherwise. The disagreement matrix is the $T \times T$ binary matrix with coefficient (s, t) equal to 1 where the two adjacency matrices disagree, and 0 otherwise. The value of Δ_{RI} is equal to the white area (where coefficients are 0) of the disagreement matrix.

227 samples, *i.e.*

$$\text{TP}(\mathcal{T}^*, \hat{\mathcal{T}}) := \{t^* \in \mathcal{T}^* \mid \exists \hat{t} \in \hat{\mathcal{T}} \text{ s.t. } |\hat{t} - t^*| < M\}. \quad (7)$$

228 Precision PREC and recall REC are then given by

$$\text{PREC}(\mathcal{T}^*, \hat{\mathcal{T}}) := |\text{TP}(\mathcal{T}^*, \hat{\mathcal{T}})| / \hat{K} \quad \text{and} \quad \text{REC}(\mathcal{T}^*, \hat{\mathcal{T}}) := |\text{TP}(\mathcal{T}^*, \hat{\mathcal{T}})| / K^*. \quad (8)$$

229 PRECISION and RECALL are well-defined (ie. between 0 and 1) if the margin M
 230 is smaller than the minimum spacing between two true change point indexes t_k^*
 231 and t_{k+1}^* . Over-segmentation of a signal causes the precision to be close to zero
 232 and the recall close to one. Under-segmentation has the opposite effect. The
 233 F1 score is the harmonic mean of precision PREC and recall REC:

$$\Delta_{\text{F1}}(\mathcal{T}^*, \hat{\mathcal{T}}) := 2 \times \frac{\text{PREC}(\mathcal{T}^*, \hat{\mathcal{T}}) \times \text{REC}(\mathcal{T}^*, \hat{\mathcal{T}})}{\text{PREC}(\mathcal{T}^*, \hat{\mathcal{T}}) + \text{REC}(\mathcal{T}^*, \hat{\mathcal{T}})}. \quad (9)$$

234 Its best value is 1 and its worse value is 0. An illustrative example is displayed
 235 on Figure 5.

236 4. Models and cost functions

237 This section presents the first defining element of change detection methods,
 238 namely the cost function. In most cases, cost functions are derived from a
 239 signal model. In the following, models and their associated cost function are
 240 organized in two categories: parametric and non-parametric, as schematically
 241 shown in Figure 6. For each model, the most general formulation is first given,

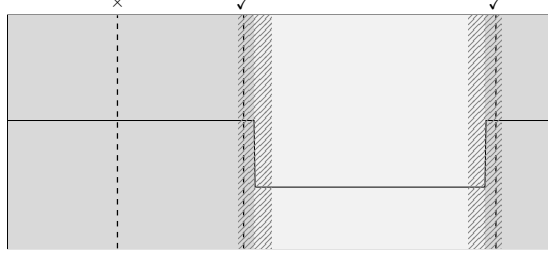


Figure 5: F1-score: Δ_{F1} . Alternating gray areas mark the segmentation \mathcal{T}^* ; dashed lines mark the segmentation $\hat{\mathcal{T}}$; dashed areas mark the allowed margin of error around true change points. Here, precision and recall are equal to $2/3$ and $2/2$ respectively, and Δ_{F1} is equal to $4/5$.

then special cases, if any, are described. A summary table of all reviewed costs can be found at the end of this section.

4.1. Parametric models

Parametric detection methods focus on changes in a finite-dimensional parameter vector. Historically, they were the first to be introduced, and remain extensively studied in the literature.

4.1.1. Maximum likelihood estimation

Maximum likelihood procedures are ubiquitous in the change point detection literature. They generalize a large number of models and cost functions, such as mean-shifts and scale shifts in normally distributed data [2, 49–51], changes in the rate parameter of Poisson distributed data [39], etc. In the general setting of maximum likelihood estimation for change detection, the observed signal $y = \{y_1, \dots, y_T\}$ is composed of independent random variables, such that

$$y_t \sim \sum_{k=0}^{K^*} f(\cdot|\theta_k) \mathbb{1}(t_k^* < t \leq t_{k+1}^*) \quad (\text{M1})$$

where the t_k^* are change point indexes, the $f(\cdot|\theta)$ are probability density functions parametrized by the vector-valued parameter θ , and the θ_k are parameter values. In other words, the signal y is modelled by iid variables with piecewise constant distribution. The parameter θ represents a quantity of interest whose value changes abruptly at the unknown instants t_k^* , which are to be estimated. Under this setting, change point detection is equivalent to maximum likelihood estimation if the sum of cost $V(\mathcal{T}, y)$ is equal to the negative log-likelihood. The corresponding cost function, denoted $c_{\text{i.i.d.}}$, is defined as follows.

Cost function 1 ($c_{\text{i.i.d.}}$). For a given parametric family of distribution densities $\{f(\cdot|\theta)|\theta \in \Theta\}$ where Θ is a compact subset of \mathbb{R}^p (for a certain p), the cost

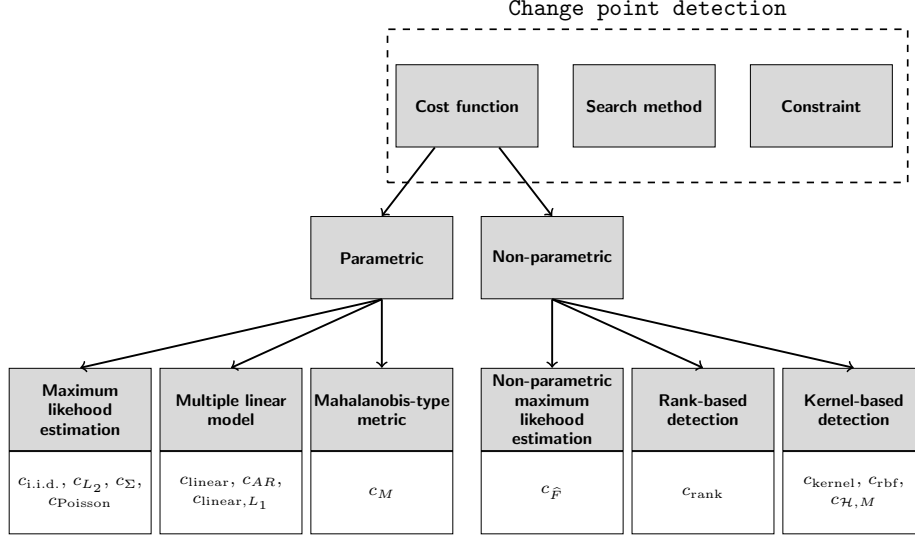


Figure 6: Typology of the cost functions described in Section 4.

function $c_{i.i.d.}$ is defined by

$$c_{i.i.d.}(y_{a..b}) := - \sup_{\theta} \sum_{t=a+1}^b \log f(y_t|\theta). \quad (C1)$$

Model M1 and the related cost function $c_{i.i.d.}$ encompasses a large number of change point methods. Note that, in this context, the family of distributions must be known before performing the detection, usually thanks to prior knowledge on the data. Historically, the Gaussian distribution was first used, to model mean-shifts [52–54] and scale shifts [39, 50, 55]. A large part of the literature then evolved towards other parametric distributions, most notably resorting to distributions from the general exponential family [15, 25, 56].

From a theoretical point of view, asymptotic consistency, as described in Definition 1, has been demonstrated, in the case of a *single change point*, first with Gaussian distribution (fixed variance), then for several specific distributions, e.g. Gaussian with mean and scale shifts [3, 6, 51, 57], discrete distributions [49], etc. The case with *multiple change points* has been tackled later. For certain distributions (e.g. Gaussian), the solutions of the change point detection problems (P1) (known number of change points) and (P2) (unknown number of change points) have been proven to be asymptotically consistent [58]. The general case of multiple change points and a generic distribution family has been addressed decades after the change detection problem has been introduced: the solution of the change point detection problem with a known number of changes and a cost function set to $c_{i.i.d.}$ is asymptotically consistent [59]. This is true if certain assumptions are satisfied: (i) the signal follows the model (M1) for a

286 distribution family that verifies some regularity assumptions (which are no dif-
 287 ferent from the assumptions needed for generic maximum likelihood estimation,
 288 without any change point) and (ii) technical assumptions on the value of the
 289 cost function on homogeneous and heterogeneous sub-signals. As an example,
 290 distributions from the exponential family satisfy those assumptions.

291 *Related cost functions.* The general model (M1) has been applied with different
 292 families of distributions. We list below three notable examples and the associ-
 293 ated cost functions: change in mean, change in mean and scale, and change in
 294 the rate parameter of count data.

- 296 • The mean-shift model is the earliest and one of the most studied model in
 297 the change point detection literature [2, 53, 60–62]. Here, the distribution
 298 is Gaussian, with fixed variance. In other words, the signal y is simply a
 299 sequence of independent normal random variables with piecewise constant
 300 mean and same variance. In this context, the cost function $c_{\text{i.i.d.}}$ becomes
 301 c_{L_2} , defined below. This cost function is also referred to as the quadratic
 302 error loss and has been applied for instance on DNA array data [15] and
 303 geology signals [6].

304 **Cost function 2** (c_{L_2}). *The cost function c_{L_2} is given by*

$$c_{L_2}(y_{a..b}) := \sum_{t=a+1}^b \|y_t - \bar{y}_{a..b}\|_2^2 \quad (\text{C2})$$

305 where $\bar{y}_{a..b}$ is the empirical mean of the sub-signal $y_{a..b}$.

- 306 • A natural extension to the mean-shift model consists in letting the variance
 307 abruptly change as well. In this context, the cost function $c_{\text{i.i.d.}}$ becomes
 308 c_Σ , defined below. This cost function can be used to detect changes in the
 309 first two moments of random (not necessarily Gaussian) variables, even
 310 though it is the Gaussian likelihood that is plugged in $c_{\text{i.i.d.}}$ [8, 49]. It
 311 has been applied for instance on stock market time series [49], biomedical
 312 data [6], and electric power consumption monitoring [63].

313 **Cost function 3** (c_Σ). *The cost function c_Σ is given by*

$$c_\Sigma(y_{a..b}) := (b - a) \log \det \hat{\Sigma}_{a..b} + \sum_{t=a+1}^b (y_t - \bar{y}_{a..b})' \hat{\Sigma}_{a..b}^{-1} (y_t - \bar{y}_{a..b}) \quad (\text{C3})$$

314 where $\bar{y}_{a..b}$ and $\hat{\Sigma}_{a..b}$ are respectively the empirical mean and the empirical
 315 covariance matrix of the sub-signal $y_{a..b}$.

- 316 • Change point detection has also been applied on count data modelled by a
 317 Poisson distribution [39, 64]. More precisely, the signal y is a sequence of

independent Poisson distributed random variables with piecewise constant rate parameter. In this context, the cost function $c_{\text{i.i.d.}}$ becomes c_{Poisson} , defined below.

Cost function 4 (c_{Poisson}). The cost function c_{Poisson} is given by

$$c_{\text{Poisson}}(y_{a..b}) := -(b - a)\bar{y}_{a..b} \log \bar{y}_{a..b} \quad (\text{C4})$$

where $\bar{y}_{a..b}$ is the empirical mean of the sub-signal $y_{a..b}$.

Remark 1. A model slightly more general than (M1) can be formulated by letting the signal samples to be dependant and the distribution function $f(\cdot|\theta)$ to change 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 [49]). The function $f(\cdot|\theta)$ is replaced in (M1) 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 only the variations of the parameter θ must be detected. Properties on the asymptotic consistency of change point estimates can be obtained in this context. We refer the reader to [49, 65] for theoretical results.

4.1.2. Piecewise linear regression

Piecewise linear models are often found, most notably in the econometrics literature, to detect so-called “structural changes” [66–68]. In this context, a linear relationship between a response variable and covariates exists, and this relationship changes abruptly at some unknown instants. The observed signal y is regarded as a univariate response variable which follows a piecewise linear model with change points located at the t_k^* :

$$\forall t, t_k^* < t \leq t_{k+1}^*, \quad y_t = x_t' u_k + z_t' v + \varepsilon_t \quad (k = 0, \dots, K^*) \quad (\text{M2})$$

where the $u_k \in \mathbb{R}^p$ and $v \in \mathbb{R}^q$ are unknown regression parameters and ε_t is noise. Under this setting, the observed signal y is the response variable and the signals $x = \{x_t\}_{t=1}^T$ and $z = \{z_t\}_{t=1}^T$ are observed covariates, respectively \mathbb{R}^p -valued and \mathbb{R}^q -valued. Note that in this section, the input signal y is therefore necessarily an univariate signal.

In the literature, Model (M2) is also known as a *partial* structural change model because the linear relationship between y and x changes abruptly, while the linear relationship between y and z remains constant. The *pure* structural change model is obtained by removing the term $z_t' v$ from (M2). The formulation (M2) generalizes several well-known models such as the autoregressive (AR) model [12, 69], multiple regressions [68, 70], etc. A more general formulation of (M2) that can accommodate a multivariate response variable y exists [71], but is more involved, from a notational standpoint.

The procedure to detect *pure* structural changes amounts to fitting a linear regression on each segment of the signal. To that end, the sum of costs is made

equal to the sum of squared residuals. The corresponding cost function, denoted c_{linear} , is defined as follows.

Cost function 5 (c_{linear}). For a signal y (response variable) and covariates x and z , the cost function c_{linear} is defined by

$$c_{\text{linear}}(y_{a..b}) := \min_{u \in \mathbb{R}^p} \sum_{t=a+1}^b (y_t - x'_t u)^2. \quad (\text{C5})$$

Detecting *partial* structural changes is more complex because the regression parameters contained in v are shared by all segments, and therefore, the associated sum of squared residuals is no longer a sum of segment costs, as in (1). Approximate procedures have been proposed to minimize the sum of residuals. For instance, one can iteratively alternate between the two following steps: keep v fixed and detect pure structural changes in $y_t - z'_t v$ (using c_{linear}), then keep the segmentation fixed and minimize over v and the u_k of each segment [72]. While finding the global minimum is not guaranteed, the authors argued, using numerical simulations, that their method “has very rapid convergence” (no more than two iterations were enough to find the minimum).

From a theoretical point of view, piecewise linear models are extensively studied in the context of change point detection by a series of important contributions [14, 66–70, 73–77]. When the number of changes is known, the most general consistency result can be found in [14]. A multivariate extension of this result has been demonstrated in [71]. As for the more difficult situation of an unknown number of changes, statistical tests have been proposed for a single change [78] and multiple changes [74]. All of those results are obtained under various sets of general assumptions on the distributions of the covariates and the noise. The most general of those sets can be found in [79]. Roughly, in addition to some technical assumptions, it imposes the processes x and z to be weakly stationary within each regime, and precludes the noise process to have a unit root.

A few articles from the literature adopt a more complex view of partial changes, in which all covariates are pooled and it is not known which ones, in (M2), belong to x (i.e. their linear relationships with the response variable y abruptly change over time) or to z (i.e. their linear relationships with the response variable y remain constant over time). Since treating partial structural changes as pure structural changes can lead to underestimation of the number of changes [80], better methods are needed. Often, segmentations are computed for all combinations of regressors in z or in x , then the optimal segmentation and combination of covariates are chosen according a statistical criterion [79]. A more recent work [80] compares a number of procedures and propose extensions, based on a finely tuned Bayesian Information Criterion (BIC).

Related cost functions. In the rich literature related to piecewise linear models, the cost function c_{linear} has been applied and extended in several different settings. Two related cost functions are listed below.

- The first one is c_{linear,L_1} , which was introduced in order to accommodate certain noise distributions with heavy tails [67, 73] and is defined as follows. Again, it is designed to detect pure structural changes.

Cost function 6 (c_{linear,L_1}). For a signal y (response variable) and covariates x and z , the cost function c_{linear,L_1} is defined by

$$c_{\text{linear},L_1}(y_{a..b}) := \min_{u \in \mathbb{R}^p} \sum_{t=a+1}^b |y_t - x'_t u - z'_t v|. \quad (\text{C6})$$

The difference between c_{linear,L_1} and c_{linear} lies in the norm used to measure errors: c_{linear,L_1} is based on a least absolute deviations criterion, while c_{linear} is based on a least squares criterion. As a result, c_{linear,L_1} is often applied on data with noise distributions with heavy tails [8, 25]. In practice, the cost function c_{linear,L_1} is computationally less efficient than the cost function c_{linear} , because the associated minimization problem (C6) has no analytical solution. Nevertheless, the cost function c_{linear,L_1} is often applied on economic and financial data [66–68]. For instance, changes in several economic parameters of the G-7 growth have been investigated using a piecewise linear model and c_{linear,L_1} [81].

- The second cost function related to c_{linear} has been introduced to deal with piecewise autoregressive signals. The autoregressive model is a popular representation of random processes, where each variable depends linearly on the previous variables. The associated cost function, denoted c_{AR} , is defined as follows.

Cost function 7 (c_{AR}). For a signal y and an order $p \geq 1$, the cost function c_{AR} is defined by

$$c_{\text{AR}}(y_{a..b}) := \min_{u \in \mathbb{R}^p} \sum_{t=a+1}^b \|y_t - x'_t u\|^2 \quad (\text{C7})$$

where $x_t := [y_{t-1}, y_{t-2}, \dots, y_{t-p}]$ is the vector of lagged samples.

The piecewise autoregressive model is a special case of the generic piecewise linear model, where the term $z'_t v$ is removed (yielding a pure structural change model) and the covariate signal x is equal to the signal of lagged samples. The resulting cost function c_{AR} is able to detect shifts in the autoregressive coefficients of a non-stationary process [21, 69]. This model has been applied on EEG/ECG time series [71], functional magnetic resonance imaging (fMRI) time series [82] and speech recognition tasks [12].

426 *4.1.3. Mahalanobis-type metric*

427 The cost function c_{L_2} (C2), adapted for mean-shift detection, can be ex-
 428 tended through the use of Mahalanobis-type seminorm. Formally, for any sym-
 429 metric positive semi-definite matrix $M \in \mathbb{R}^{d \times d}$, the associated seminorm $\|\cdot\|_M$
 430 is given by:

$$\|y_t\|_M^2 := y_t' M y_t \quad (10)$$

431 for any sample y_t . The resulting cost function c_M is defined as follows.

432 **Cost function 8 (c_M).** *The cost function c_M , parametrized by a symmetric*
 433 *positive semi-definite matrix $M \in \mathbb{R}^{d \times d}$, is given by*

$$c_M(y_{a..b}) := \sum_{t=a+1}^b \|y_t - \bar{y}_{a..b}\|_M^2 \quad (C8)$$

434 where $\bar{y}_{a..b}$ is the empirical mean of the sub-signal $y_{a..b}$.

435 Intuitively, measuring distances with the seminorm $\|\cdot\|_M$ is equivalent to apply-
 436 ing a linear transformation on the data and using the regular (Euclidean) norm
 437 $\|\cdot\|$. Indeed, decomposing the matrix $M = U'U$ yields:

$$\|y_t - y_s\|_M^2 = \|Uy_t - Uy_s\|^2. \quad (11)$$

438 Originally, the metric matrix M was set equal to the inverse of the covariance
 439 matrix, yielding the Mahalanobis metric [83], ie

$$M = \hat{\Sigma}^{-1} \quad (12)$$

440 where $\hat{\Sigma}$ is the empirical covariance matrix of the signal y . By using c_M , shifts
 441 in the mean of the transformed signal can be detected. In practice, the trans-
 442 formation U (or equivalently, the matrix M) is chosen to highlight relevant
 443 changes. This cost function generalizes all linear transformations of the data
 444 samples. In the context of change point detection, most of the transformations
 445 are unsupervised, for instance principal component analysis or linear discrim-
 446 inant analysis [84]. Supervised strategies are more rarely found, even though
 447 there exist numerous methods to learn a task-specific matrix M in the context
 448 of supervised classification [84–86]. Those strategies fall under the umbrella
 449 of metric learning algorithms. In the change point detection literature, there is
 450 only one work that proposes a supervised procedure to calibrate a metric matrix
 451 M [30]. In this contribution, the authors use a training set of annotated sig-
 452 nals (meaning that an expert has provided the change point locations) to learn
 453 M iteratively. Roughly, at each step, a new matrix M is generated in order
 454 to improve change point detection accuracy on the training signals. However,
 455 using the cost function c_M is not adapted to certain applications, where a linear
 456 treatment of the data is insufficient. In that situation, a well-chosen non-linear
 457 transformation of the data samples must be applied beforehand [30].

458 4.2. Non-parametric models

459 When the assumptions of parametric models are not adapted to the data
 460 at hand, non-parametric change point detection methods can be more robust.
 461 Three major approaches are presented here, each based on different non-parametric
 462 statistics, such as the empirical cumulative distribution function, rank statistics
 463 and kernel estimation.

464 *Signal model.* Assume that the observed signal $y = \{y_1, \dots, y_T\}$ is composed
 465 of independent random variables, such that

$$y_t \sim \sum_{k=0}^{K^*} F_k \mathbb{1}(t_k^* < t \leq t_{k+1}^*) \quad (\text{M3})$$

466 where the t_k^* are change point indexes and the F_k are cumulative distribution
 467 functions (c.d.f.), not necessarily parametric as in (M1). Under this setting, the
 468 sub-signal $y_{t_k^* \dots t_{k+1}^*}$, bounded by two change points, is composed of iid variables
 469 with c.d.f. F_k . When the F_k belong to a known parametric distribution fam-
 470 ily, change point detection is performed with the MLE approach described in
 471 Section 4.1.1, which consists in applying the cost function $c_{\text{i.i.d.}}$. However, this
 472 approach is not possible when the distribution family is either non-parametric
 473 or not known beforehand.

474 4.2.1. Non-parametric maximum likelihood

475 The first non-parametric cost function example, denoted $c_{\hat{F}}$, has been intro-
 476 duced for the *single* change point detection problem in [87] and extended for
 477 *multiple* change points in [88]. It relies on the empirical cumulative distribu-
 478 tion function, estimated on sub-signals. Formally, the signal is assumed to be
 479 univariate (ie $d = 1$) and the empirical cdf on the sub-signal $y_{a..b}$ is given by [89]

$$\forall u \in \mathbb{R}, \quad \hat{F}_{a..b}(u) := \frac{1}{b-a} \left[\sum_{t=a+1}^b \mathbb{1}(y_t < u) + 0.5 \times \mathbb{1}(y_t = u) \right]. \quad (13)$$

480 In order to derive a log-likelihood function that does not depend on the proba-
 481 bility distribution of the data, ie the $f(\cdot|\theta_k)$, the authors use the following fact:
 482 for a fixed $u \in \mathbb{R}$, the empirical cdf \hat{F} of n iid random variables, distributed from
 483 a certain cdf F is such that $n\hat{F}(u) \sim \text{Binomial}(n, F(u))$ [88]. This observation,
 484 combined with careful summation over u , allows a distribution-free maximum
 485 likelihood estimation. The resulting cost function $c_{\hat{F}}$ is defined as follows. Inter-
 486 estingly, this strategy was first introduced to design non-parametric two-sample
 487 statistical tests, which were experimentally shown to be more powerful than
 488 classical tests such as Kolmogorov-Smirnov and Cramr-von Mises [87, 90].

489 **Cost function 9 ($c_{\hat{F}}$).** The cost function $c_{\hat{F}}$ is given by

$$c_{\hat{F}}(y_{a..b}) := -(b-a) \sum_{u=1}^T \frac{\hat{F}_{a..b}(u) \log \hat{F}_{a..b}(u) + (1 - \hat{F}_{a..b}(u)) \log(1 - \hat{F}_{a..b}(u))}{(u-0.5)(T-u+0.5)} \quad (\text{C9})$$

490 where the empirical cdf $\hat{F}_{a..b}$ is defined by (13).

491 From a theoretical point of view, asymptotic consistency of change point es-
 492 timates is verified, when the number of change points is either known or un-
 493 known [88]. However, solving either one of the detection problems can be com-
 494 putationally intensive, because calculating the value of the cost function $c_{\hat{F}}$ on
 495 one sub-signal requires to sum T terms, where T is the signal length. As a result,
 496 the total complexity of change point detection is of the order of $\mathcal{O}(T^3)$ [88]. To
 497 cope with this computational burden, several preliminary steps are proposed.
 498 For instance, irrelevant change point indexes can be removed before performing
 499 the detection, thanks to a screening step [88]. Also, the cost function $c_{\hat{F}}$ can
 500 be approximated, by summing, in (C9), over a few (carefully chosen) terms,
 501 instead of T terms originally [89]. Thanks to those implementation techniques,
 502 the cost function $c_{\hat{F}}$ has been applied on DNA sequences [88] and heart-rate
 503 monitoring signals [89].

504 4.2.2. Rank-based detection

505 In statistical inference, a popular strategy to derive distribution-free statis-
 506 tics is to replaced the data samples by their ranks within the set of pooled
 507 observations [28, 91, 92]. In the context of change point detection, this strategy
 508 has first been applied to detect a *single* change point [28, 29], and then has been
 509 extended by [93] to find *multiple* change points. The associated cost function,
 510 denoted c_{rank} , is defined as follows. Formally, it relies on the centered \mathbb{R}^d -valued
 511 “rank signal” $r = \{r_t\}_{t=1}^T$, given by

$$r_{t,j} := \sum_{s=1}^T \mathbf{1}(y_{s,j} \leq y_{t,j}) - \frac{T+1}{2}, \quad \forall 1 \leq t \leq T, \forall 1 \leq j \leq d. \quad (14)$$

512 In other words, $r_{t,j}$ is the (centered) rank of the j^{th} coordinate of the t^{th} sample,
 513 ie $y_{t,j}$, among the $\{y_{1,j}, y_{2,j}, \dots, y_{T,j}\}$.

514 **Cost function 10 (c_{rank}).** The cost function c_{rank} is given by

$$c_{\text{rank}}(y_{a..b}) := -(b-a) \bar{r}'_{a..b} \hat{\Sigma}_r^{-1} \bar{r}_{a..b} \quad (\text{C10})$$

515 where the signal r is defined in (14), $\bar{r}_{a..b}$ is the empirical mean of the sub-signal
 516 $\{r_t\}_{t=a+1}^b$ and $\hat{\Sigma}_r \in \mathbb{R}^{d \times d}$ is the following matrix

$$\hat{\Sigma}_r := \frac{1}{T} \sum_{t=1}^T (r_t + \mathbf{1}_d/2) (r_t + \mathbf{1}_d/2)' \quad (15)$$

with $\mathbf{1}_d \in \mathbb{R}^d$ the all-ones vector. Intuitively, c_{rank} measures changes in the joint behaviour of the marginal rank statistics of each coordinate, which are contained in r . One of the advantages of this cost function is that it is invariant under any monotonic transformation of the data. Several well-known statistical hypothesis testing procedures are based on this scheme, for instance the Wilcoxon-Mann-Whitney test [94], the Friedman test [95], the Kruskal-Wallis test [96], and several others [91, 92]. From a computational point of view, two steps must be performed before the change point detection: the calculation of the rank statistics, in $\mathcal{O}(dT \log T)$ operations, and the calculation of the matrix $\hat{\Sigma}_r$, in $\mathcal{O}(d^2T + d^3)$ operations. The resulting algorithm has been applied on DNA sequences [93] and network traffic data [28, 29].

4.2.3. Kernel-based detection

A kernel-based method has been proposed by [97] to perform change point detection in a non-parametric setting. To that end, the original signal y is mapped onto a reproducing Hilbert space (rkhs) \mathcal{H} associated with a user-defined kernel function $k(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$. The mapping function $\phi : \mathbb{R}^d \rightarrow \mathcal{H}$ onto this rkhs is implicitly defined by $\phi(y_t) = k(y_t, \cdot) \in \mathcal{H}$, resulting in the following inner-product and norm:

$$\langle \phi(y_s) | \phi(y_t) \rangle_{\mathcal{H}} = k(y_s, y_t) \quad \text{and} \quad \|\phi(y_t)\|_{\mathcal{H}}^2 = k(y_t, y_t) \quad (16)$$

for any samples $y_s, y_t \in \mathbb{R}^d$. The associated cost function, denoted c_{kernel} , is defined as follows. This kernel-based mapping is central to many machine learning developments such as support vector machine or clustering [98, 99].

Cost function 11 (c_{kernel}). For a given kernel function $k(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$, the cost function c_{kernel} is given by

$$c_{\text{kernel}}(y_{a..b}) := \sum_{t=a+1}^b \|\phi(y_t) - \bar{\mu}_{a..b}\|_{\mathcal{H}}^2 \quad (\text{C11})$$

where $\bar{\mu}_{a..b} \in \mathcal{H}$ is the empirical mean of the embedded signal $\{\phi(y_t)\}_{t=a+1}^b$ and $\|\cdot\|_{\mathcal{H}}$ is defined in (16).

Remark 2 (Computing the cost function). Thanks to the well-known “kernel trick”, the explicit computation of the mapped data samples $\phi(y_t)$ is not required to calculate the cost function value [100]. Indeed, after simple algebraic manipulations, $c_{\text{kernel}}(y_{a..b})$ can be rewritten as follows:

$$c_{\text{kernel}}(y_{a..b}) = \sum_{t=a+1}^b k(y_t, y_t) - \frac{1}{b-a} \sum_{s,t=a+1}^b k(y_s, y_t). \quad (17)$$

Remark 3 (Intuition behind the cost function). Intuitively, the cost function c_{kernel} is able to detect mean-shifts in the transformed signal $\{\phi(y_t)\}_t$. Its

use is motivated in the context of Model M3 by the fact that, under certain conditions on the kernel function, changes in the probability distribution coincide with mean-shifts in the transformed signal. This connection has been investigated in several works on kernel methods [98, 99, 101, 102].

Formally, let \mathbb{P} denote a Borel probability measure defined over \mathbb{R}^d and assume that $k(\cdot, \cdot)$ is measurable and $\mathbb{E}_{X \sim \mathbb{P}}[k(X, X)] < +\infty$.

Then there exists a unique element $\mu_{\mathbb{P}} \in \mathcal{H}$ [99], called the mean embedding (of \mathbb{P}), such that

$$\mu_{\mathbb{P}} = \mathbb{E}_{X \sim \mathbb{P}} [\phi(X)]. \quad (18)$$

In addition, under certain conditions on the kernel $k(\cdot, \cdot)$, the mapping $\mathbb{P} \mapsto \mu_{\mathbb{P}}$ is injective (in which case the kernel is said to be characteristic), meaning that

$$\forall \mathbb{P}, \mathbb{Q}, \quad \mu_{\mathbb{P}} = \mu_{\mathbb{Q}} \Rightarrow \mathbb{P} = \mathbb{Q}, \quad (19)$$

where \mathbb{Q} denotes a Borel probability measure defined over \mathbb{R}^d .

In order to determine if a kernel is characteristic (and therefore, useful for change point detection), several conditions can be found in the literature [98, 99, 101]. For instance, if a 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) and verifies a condition on its Fourier transform, then it is characteristic [101]. This condition is verified by the commonly used Gaussian kernel. As a consequence, two transformed samples $\phi(y_s)$ and $\phi(y_t)$ are distributed around the same mean value if they belong to the same regime, and around different mean-values if they each belong to two consecutive regimes. To put it another way, a signal that follows (M3) is mapped by $\phi(\cdot)$ to a random signal with piecewise constant mean.

From a theoretical point of view, asymptotic consistency of the change point estimates has been demonstrated for both a known and unknown number of change points in the recent work of [103]. This result, as well as an important oracle inequality on the sum of cost $V(\mathcal{T})$ [104], also holds in a non-asymptotic setting. In addition, kernel change point detection was experimentally shown to be competitive in many different settings, in an unsupervised manner and with very few parameters to manually calibrate. For instance, the cost function c_{kernel} was applied on the Brain-Computer Interface (BCI) data set [97], on a video time series segmentation task [104], DNA sequences [100] and emotion recognition [105].

Related cost functions. The cost function c_{kernel} can be combined with any kernel to accommodate various types of data (not just \mathbb{R}^d -valued signals). Notable examples of kernel functions include [102]:

- The linear kernel $k(x, y) = \langle x | y \rangle$ with $x, y \in \mathbb{R}^d$.
- The polynomial kernel $k(x, y) = (\langle x | y \rangle + C)^{\text{deg}}$ with $x, y \in \mathbb{R}^d$, and C and deg are parameters.

- The Gaussian kernel $k(x, y) = \exp(-\gamma \|x - y\|^2)$ with $x, y \in \mathbb{R}^d$ and $\gamma > 0$ is the so-called bandwidth parameter.
- The χ^2 -kernel $k(x, y) = \exp(-\gamma \sum_i [(x_i - y_i)^2 / (x_i + y_i)])$ with $\gamma \in \mathbb{R}$ a parameter. It is often used for histogram data.

Arguably, the most commonly used kernels for numerical data are the linear kernel and the Gaussian kernel. When combined with the linear kernel, the cost function c_{kernel} is formally equivalent to c_{L_2} . As for the Gaussian kernel, the associated cost function, denoted c_{rbf} , is defined as follows.

Cost function 12 (c_{rbf}). *The cost function c_{rbf} is given by*

$$c_{\text{rbf}}(y_{a..b}) := (b - a) - \frac{1}{b - a} \sum_{s,t=a+1}^b \exp(-\gamma \|y_s - y_t\|^2) \quad (\text{C12})$$

where $\gamma > 0$ is the so-called bandwidth parameter.

The parametric cost function c_M (based on a Mahalanobis-type norm) can be extended to the non-parametric setting through the use of a kernel. Formally, the Mahalanobis-type norm $\|\cdot\|_{\mathcal{H},M}$ in the feature space \mathcal{H} is defined by

$$\|\phi(y_s) - \phi(y_t)\|_{\mathcal{H},M}^2 = (\phi(y_s) - \phi(y_t))' M (\phi(y_s) - \phi(y_t)) \quad (20)$$

where M is a (possibly infinite dimensional) symmetric positive semi-definite matrix defined on \mathcal{H} .

The so-called metric matrix M is often chosen to enforce distance constraints, for instance, have (relatively) small distances between samples labelled as similar and (relatively) large distances between samples labelled as dissimilar [85, 86, 106]. Thus defined (20), the seminorm $\|\cdot\|_{\mathcal{H},M}$ is unpractical to compute because of the infinite dimension of arbitrary M . However, there exist procedures (particularly in the metric learning literature) [106, 107] that learn an appropriate metric matrix, which can be fully represented by the constrained training samples. In such a setting, computing $\|\cdot\|_{\mathcal{H},M}$ amounts to evaluating $k(\cdot, \cdot)$ on the set of training samples.

The associated cost function, denoted $c_{\mathcal{H},M}$, is defined below. Intuitively, using $c_{\mathcal{H},M}$ instead of c_M introduces a non-linear treatment of the data samples.

Cost function 13 ($c_{\mathcal{H},M}$). *For a given rkhs \mathcal{H} , implicitly determined by a kernel function $k(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$, and M a symmetric positive semi-definite matrix defined on \mathcal{H} , the cost function $c_{\mathcal{H},M}$ is given by*

$$c_{\mathcal{H},M}(y_{a..b}) := \sum_{t=a+1}^b \|\phi(y_t) - \bar{\mu}_{a..b}\|_{\mathcal{H},M}^2 \quad (\text{C13})$$

where $\bar{\mu}_{a..b}$ is the empirical mean of the transformed sub-signal $\{\phi(y_t)\}_{t=a+1}^b$ and $\|\cdot\|_{\mathcal{H},M}$ is defined in (20).

618 *4.3. Summary table*

619 Reviewed cost functions (parametric and non-parametric) are summarized
620 in Table 1. For each cost, the name, expression and parameters of interest are
621 given.

Name	$c(y_{a..b})$	Parameters
$c_{i.i.d.}$ (C1)	$-\sup_{\theta} \sum_{t=a+1}^b \log f(y_t \theta)$	θ : changing parameter; density function: $f(\cdot \theta)$
c_{L_2} (C2)	$\sum_{t=a+1}^b \ y_t - \bar{y}_{a..b}\ _2^2$	$\bar{y}_{a..b}$: empirical mean of $y_{a..b}$
c_{Σ} (C3)	$(b-a) \log \det \hat{\Sigma}_{a..b} + \sum_{t=a+1}^b (y_t - \bar{y}_{a..b})' \hat{\Sigma}_{a..b}^{-1} (y_t - \bar{y}_{a..b})$	$\hat{\Sigma}_{a..b}$: empirical covariance of $y_{a..b}$
c_{Poisson} (C4)	$-(b-a) \bar{y}_{a..b} \log \bar{y}_{a..b}$	$\bar{y}_{a..b}$: empirical mean of $y_{a..b}$
c_{linear} (C5)	$\min_{u \in \mathbb{R}^p} \sum_{t=a+1}^b (y_t - x_t' u)^2$	$x_t \in \mathbb{R}^p$: covariates
c_{linear, L_1} (C6)	$\min_{u \in \mathbb{R}^p} \sum_{t=a+1}^b y_t - x_t' u $	$x_t \in \mathbb{R}^p$: covariates
c_{AR} (C7)	$\min_{u \in \mathbb{R}^p} \sum_{t=a+1}^b (y_t - x_t' u)^2$	$x_t = [y_{t-1}, y_{t-2}, \dots, y_{t-p}]$: lagged samples
c_M (C8)	$\sum_{t=a+1}^b \ y_t - \bar{y}_{a..b}\ _M^2$	$M \in \mathbb{R}^{d \times d}$: positive semi-definite matrix
$c_{\hat{F}}$ (C9)	$-(b-a) \sum_{u=1}^T \frac{\hat{F}_{a..b}(u) \log \hat{F}_{a..b}(u) + (1 - \hat{F}_{a..b}(u)) \log(1 - \hat{F}_{a..b}(u))}{(u-0.5)(T-u+0.5)}$	\hat{F} : empirical c.d.f. (13)
c_{rank} (C10)	$-(b-a) \bar{r}_{a..b}' \hat{\Sigma}_r^{-1} \bar{r}_{a..b}$	r : rank signal (14); $\hat{\Sigma}_r$: empirical covariance of r (15)
c_{kernel} (C11)	$\sum_{t=a+1}^b k(y_t, y_t) - \frac{1}{b-a} \sum_{s,t=a+1}^b k(y_s, y_t)$	$k(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}$: kernel function
c_{rbf} (C12)	$(b-a) - \frac{1}{b-a} \sum_{s,t=a+1}^b \exp(-\gamma \ y_s - y_t\ ^2)$	$\gamma > 0$: bandwidth parameter
$c_{\mathcal{H}, M}$ (C13)	$\sum_{t=a+1}^b \ y_t - \bar{y}_{a..b}\ _{\mathcal{H}, M}^2$	M : positive semi-definite matrix (in the feature space \mathcal{H})

Table 1: Summary of reviewed cost functions

622 **5. Search methods**

623 This section presents the second defining element of change detection meth-
624 ods, namely the search method. Reviewed search methods are organized in
625 two general categories, as shown on Figure 7: optimal methods, that yield the
626 exact solution to the discrete optimization of (P1) and (P2), and the approx-
627 imate methods, that yield an approximate solution. Described algorithms can
628 be combined with cost functions from Section 4. Note that, depending on the
629 chosen cost function, the computational complexity of the complete algorithm
630 changes. As a consequence, in the following, complexity analysis is done with

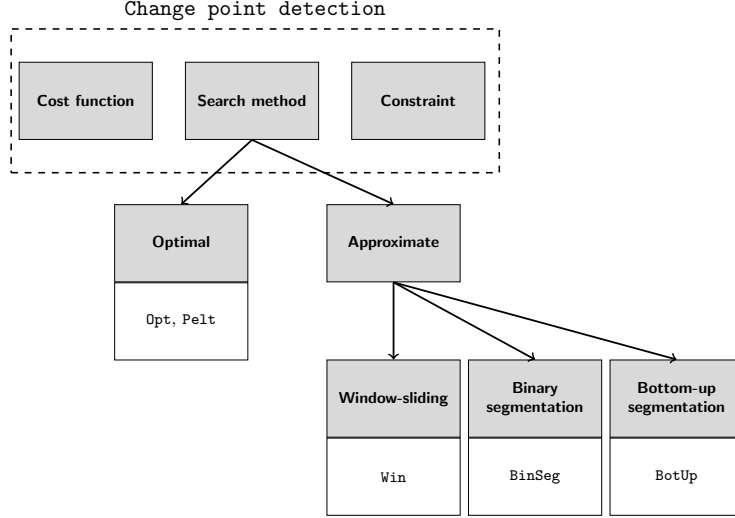


Figure 7: Typology of the search methods described in Section 5.

the assumption that applying the cost function on a sub-signal requires $\mathcal{O}(1)$ operations. Also, the practical implementations of the most important algorithms are given in pseudocode.

5.1. Optimal detection

Optimal detection methods find the exact solutions of Problem 1 (P1) and Problem 2 (P2). A naive approach consists in enumerating all possible segmentations of a signal, and returning the one that minimizes the objective function. However, for (P1), minimization is carried out over the set $\{\mathcal{T} \text{ s.t. } |\mathcal{T}| = K\}$ (which contains $\binom{T-1}{K-1}$ elements), and for (P2), over the set $\{\mathcal{T} \text{ s.t. } 1 \leq |\mathcal{T}| < T\}$ (which contains $\sum_{K=1}^{T-1} \binom{T-1}{K-1}$ elements). This makes exhaustive enumeration impractical, in both situations. We describe in this section two major approaches to efficiently find the exact solutions of (P1) and (P2).

5.1.1. Solution to Problem 1 (P1): *Opt*

In (P1), the number of change points to detect is fixed to a certain $K \geq 1$. The optimal solution to this problem can be computed efficiently, thanks to a method based on dynamic programming. The algorithm, denoted *Opt*, relies on the additive nature of the objective function $V(\cdot)$ to recursively solve sub-

648 problems. Precisely, **Opt** is based on the following observation:

$$\begin{aligned}
\min_{|\mathcal{T}|=K} V(\mathcal{T}, y = y_{0..T}) &= \min_{0=t_0 < t_1 < \dots < t_K < t_{K+1}=T} \sum_{k=0}^K c(y_{t_k..t_{k+1}}) \\
&= \min_{t \leq T-K} \left[c(y_{0..t}) + \min_{t=t_0 < t_1 < \dots < t_{K-1} < t_K=T} \sum_{k=0}^{K-1} c(y_{t_k..t_{k+1}}) \right] \\
&= \min_{t \leq T-K} \left[c(y_{0..t}) + \min_{|\mathcal{T}|=K-1} V(\mathcal{T}, y_{t..T}) \right]
\end{aligned} \tag{21}$$

649 Intuitively, Equation 21 means that the first change point of the optimal seg-
650 mentation is easily computed if the optimal partitions with $K-1$ elements of all
651 sub-signals $y_{t..T}$ are known. The complete segmentation is then computed by
652 recursively applying this observation. This strategy, described in detail in Al-
653 gorithm 1, has a complexity of the order of $\mathcal{O}(KT^2)$ [72, 108]. Historically, **Opt**
654 was introduced for a non-related problem [109] and later applied to change point
655 detection, in many different contexts, such as EEG recordings [65, 110], DNA
656 sequences [100, 111], tree growth monitoring [20], financial time-series [49, 76],
657 radar waveforms [112], etc.

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**
 $s \leftarrow L(k)$
 $t^* \leftarrow \operatorname{argmin}_{k-1 \leq t < s} C_{k-1}(1, t) + C_1(t+1, s)$
 $L(k-1) \leftarrow t^*$
 $k \leftarrow k-1$
end while
Remove T from L
Output: set L of estimated breakpoint indexes.

658 *Related search methods..* Several extensions of `Opt` have been proposed in the
659 literature. The proposed methods still find the exact solution to (P1).

- 660 - The first extension is the “forward dynamic programming” algorithm [20].
661 Contrary to `Opt`, which returns a single partition, the “forward dynamic
662 programming” algorithm computes the top L ($L \geq 1$) most probable par-
663 titions (ie with lowest sum of costs). The resulting computational com-
664 plexity is $\mathcal{O}(LKT^2)$ where L is the number of computed partitions. This
665 method is designed as a diagnostic tool: change points present in many of
666 the top partitions are considered very likely, while change points present
667 in only a few of the top partitions might not be as relevant. Thanks to
668 “forward dynamic programming”, insignificant change points are trimmed
669 and overestimation of the number of change point is corrected [20], at the
670 expense of a higher computational burden. It is applied on tree growth
671 monitoring time series [20] that are relatively short with around a hundred
672 samples.
- 673 - The “pruned optimal dynamic programming” procedure [111] is an exten-
674 sion of `Opt` that relies on a pruning rule to discard indexes that can never
675 be change points. Thanks to this trick, the set of potential change point
676 indexes is reduced. All described cost functions can be plugged into this
677 method. As a result, longer signals can be handled, for instance long array-
678 based DNA copy number data (up to 10^6 samples, with the quadratic error
679 cost function) [111]. However, worst case complexity remains of the order
680 of $\mathcal{O}(KT^2)$.

681 5.1.2. Solution to Problem 2 (P2): `Pelt`

682 In (P2), the number of changes point is unknown, and the objective function
683 to minimize is the penalized sum of costs. A naive approach consists in applying
684 `Opt` for $K = 1, \dots, K_{\max}$ for a sufficiently large K_{\max} , then choosing among the
685 computed segmentations the one that minimizes the penalized problem. This
686 would prove computational cumbersome because of the quadratic complexity
687 of the resolution method `Opt`. Fortunately a faster method exists for a general
688 class of penalty functions, namely linear penalties. Formally, linear penalties
689 are linear functions of the number of change points, meaning that

$$\text{pen}(\mathcal{T}) = \beta|\mathcal{T}| \quad (22)$$

where $\beta > 0$ is a smoothing parameter. (More details on such penalties can
be found in Section 6.1.) The algorithm `Pelt` (for “Pruned Exact Linear
Time”) [113] was introduced to find the exact solution of (P2), when the penalty
is linear (22). 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. Precisely, for two indexes t and s ($t < s < T$), the pruning rule

is given by:

$$\text{if } \left[\min_{\mathcal{T}} V(\mathcal{T}, y_{0..t}) + \beta|\mathcal{T}| \right] + c(y_{t..s}) \geq \left[\min_{\mathcal{T}} V(\mathcal{T}, y_{0..s}) + \beta|\mathcal{T}| \right] \quad \text{holds,}$$

then t cannot be the last change point prior to T . (23)

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 $\mathcal{O}(T)$. The detailed algorithm can be found in Algorithm 2. An extension of **Pelt** is described in [9] to solve the linearly penalized change point detection for a range of smoothing parameter values $[\beta_{\min}, \beta_{\max}]$. **Pelt** has been applied on DNA sequences [16, 17], physiological signals [89], and oceanographic data [113].

Algorithm 2 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 $\chi \leftarrow \{0\}$. \triangleright Admissible indexes.
for $t = 1, \dots, T$ **do**
 $\hat{t} \leftarrow \operatorname{argmin}_{s \in \chi} [Z[s] + c(y_{s..t}) + \beta]$.
 $Z[t] \leftarrow [Z[\hat{t}] + c(y_{\hat{t}..t}) + \beta]$
 $L[t] \leftarrow L[\hat{t}] \cup \{\hat{t}\}$.
 $\chi \leftarrow \{s \in \chi : Z[s] + c(y_{s..t}) \leq Z[t]\} \cup \{t\}$
end for
Output: set $L[T]$ of estimated breakpoint indexes.

5.2. *Approximate detection*

When the computational complexity of optimal methods is too great for the application at hand, one can resort to approximate methods. In this section, we describe three major types of approximate segmentation algorithms, namely window-based methods, binary segmentation and bottom-up segmentation. All described procedures fall into the category of sequential detection approaches, meaning that they return a single change point estimate $\hat{t}^{(k)}$ ($1 \leq \hat{t}^{(k)} < T$) at the k -th iteration. (In the following, the subscript $\cdot^{(k)}$ refers to the k -th iteration of a sequential algorithm.) Such methods can be used to solve (approximately) either (P1) or (P2). Indeed, if the number K^* of changes is known, K^* iterations of a sequential algorithm are enough to retrieve a segmentation with the correct number of changes. If K^* is unknown, the sequential algorithm is run until an appropriate stopping criterion is met.

5.2.1. *Window sliding*

The window-sliding algorithm, denoted **Win**, is a fast approximate alternative to optimal methods. It consists in computing the discrepancy between two

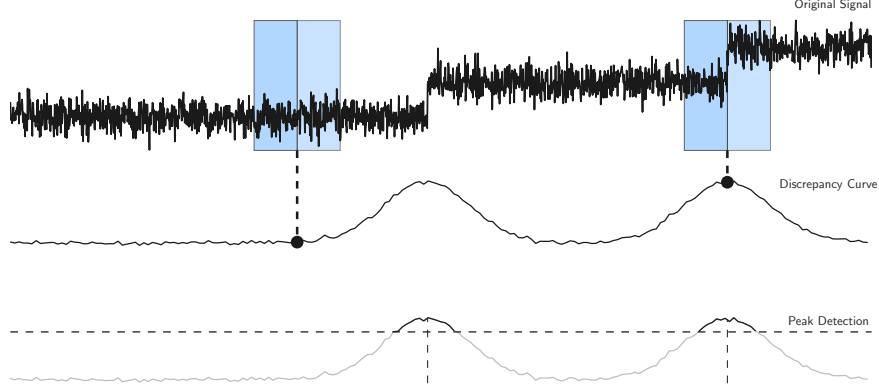


Figure 8: Schematic view of **Win**

713 adjacent windows that slide along the signal y . For a given cost function $c(\cdot)$,
 714 this discrepancy between two sub-signals is given by

$$d(y_{a..t}, y_{t..b}) = c(y_{a..b}) - c(y_{a..t}) - c(y_{t..b}) \quad (1 \leq a < t < b \leq T). \quad (24)$$

715 When the two windows cover dissimilar segments, the discrepancy reaches large
 716 values, resulting in a peak. In other other words, for each index t , **Win** measures
 717 the discrepancy between the immediate past (“left window”) and the immedi-
 718 ate future (“right window”). Once the complete discrepancy curve has been
 719 computed, a peak search procedure is performed to find change point indexes.
 720 The complete **Win** algorithm is given in Algorithm 3 and a schematic view is
 721 displayed on Figure 8. The main benefits of **Win** are its low complexity (linear
 722 in the number of samples) and ease of implementation.

Algorithm 3 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.

723 In the literature, the discrepancy measure $d(\cdot, \cdot)$ is often derived from a two-
 724 sample statistical test (see Remark 4), and not from a cost function, as in (24).
 725 However, the two standpoints are generally equivalent: for instance, using c_{L_2} ,
 726 $c_{\text{i.i.d.}}$ or c_{kernel} is respectively equivalent to applying a Student t-test [3], a gener-
 727 alized likelihood ratio (GLR) [6] test and a kernel Maximum Mean Discrepancy
 728 (MMD) test [99]. As a consequence, practitioners can capitalize on the vast
 729 body of work in the field of statistical tests to obtain asymptotic distributions
 730 for the discrepancy measure [28, 29, 99, 114], and sensible calibration strate-
 731 gies for important parameters of `Win` (such as the window size or the peak
 732 search procedure). `Win` has been applied in numerous contexts: for instance,
 733 on biological signals [11, 115–118], on network data [28, 29], on speech time
 734 series [10, 11, 119] and on financial time series [3, 120, 121]. It should be noted
 735 that certain window-based detection methods in the literature rely on a discrep-
 736 ancy measure which is not related to a cost function, as in (24) [11, 122–124].
 737 As a result, those methods, initially introduced in the online detection setting,
 738 cannot be extended to work with optimal algorithms (`Opt`, `Pelt`).

739 **Remark 4 (Two-sample test).** *A two-sample test (or homogeneity test) is*
 740 *a statistical hypothesis testing procedure designed to assess whether two popu-*
 741 *lations of samples are identical in distribution. Formally, consider two sets of*
 742 *iid \mathbb{R}^d -valued random samples $\{x_t\}_t$ and $\{z_t\}_t$. Denote by \mathbb{P}_x the distribution*
 743 *function of the x_t and by \mathbb{P}_z , the distribution function of the z_t . A two-sample*
 744 *test procedure compares the two following hypotheses:*

$$\begin{aligned}
 H_0 : \quad & \mathbb{P}_x = \mathbb{P}_z \\
 H_1 : \quad & \mathbb{P}_x \neq \mathbb{P}_z.
 \end{aligned}
 \tag{25}$$

745 A general approach is to consider a probability (pseudo)-metric $d(\cdot, \cdot)$ on the
 746 space of probability distributions on \mathbb{R}^d . Well-known examples of such a met-
 747 ric include the Kullback-Leibler divergence, the Kolmogorov-Smirnov distance,
 748 the Maximum Mean Discrepancy (MMD), etc. Observe that, under the null
 749 hypothesis, $d(\mathbb{P}_x, \mathbb{P}_z) = 0$. The testing procedure consists in computing the em-
 750 pirical estimates $\hat{\mathbb{P}}_x$ and $\hat{\mathbb{P}}_z$ and rejecting H_0 for “large” values of the statistics
 751 $d(\hat{\mathbb{P}}_x, \hat{\mathbb{P}}_z)$. This general formulation relies on a consistent estimation of arbi-
 752 trary distributions from a finite number of samples. In the parametric setting,
 753 additional assumptions are made on the distribution functions: for instance,
 754 Gaussian assumption [3, 6, 114], exponential family assumption [15, 125], etc.
 755 In the non-parametric setting, the distributions are only assumed to be con-
 756 tinuous. They are not directly estimated; instead, the statistics $d(\hat{\mathbb{P}}_x, \hat{\mathbb{P}}_z)$ are
 757 computed [11, 91, 99, 123].

758 In the context of single change point detection, the two-sample test setting is
 759 adapted to assess whether a distribution change has occurred at some instant
 760 in the input signal. Practically, for a given index t , the homogeneity test is
 761 performed on the two populations $\{y_s\}_{s \leq t}$ and $\{y_s\}_{s > t}$. The estimated change

point location is given by

$$\hat{t} = \operatorname{argmax}_t d(\hat{\mathbb{P}}_{\bullet \leq t}, \hat{\mathbb{P}}_{\bullet > t}) \quad (26)$$

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

5.2.2. Binary segmentation

Binary segmentation, denoted **BinSeg**, is a well-known alternative to optimal methods [53], because it is conceptually simple and easy to implement [6, 113, 126]. **BinSeg** is a greedy sequential algorithm, outlined as follows. The first change point estimate $\hat{t}^{(1)}$ is given by

$$\hat{t}^{(1)} := \operatorname{argmin}_{1 \leq t < T-1} \underbrace{c(y_{0..t}) + c(y_{t..T})}_{V(\mathcal{T}=\{t\})}. \quad (27)$$

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 a stopping criterion is met. A schematic view of the algorithm is displayed on Figure 9 and an implementation is given in Algorithm 4. The complexity of **BinSeg** is of the order of $\mathcal{O}(T \log T)$. This low complexity comes at the expense of optimality: in general, **BinSeg**’s output is only an approximation of the optimal solution. As argued in [113, 127], 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 [8]. Applications of **BinSeg** range from financial time series [6, 7, 114, 127, 128] to context recognition for mobile devices [129] and array-based DNA copy number data [19, 126, 130].

Related search methods. Several extensions of **BinSeg** have been proposed to improve detection accuracy.

- Circular binary segmentation [126] 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 has been combined with c_{L_2} C2, to detect changes in the mean of array-based DNA copy number data [126, 131, 132]. A faster version of the original algorithm is described in [133].

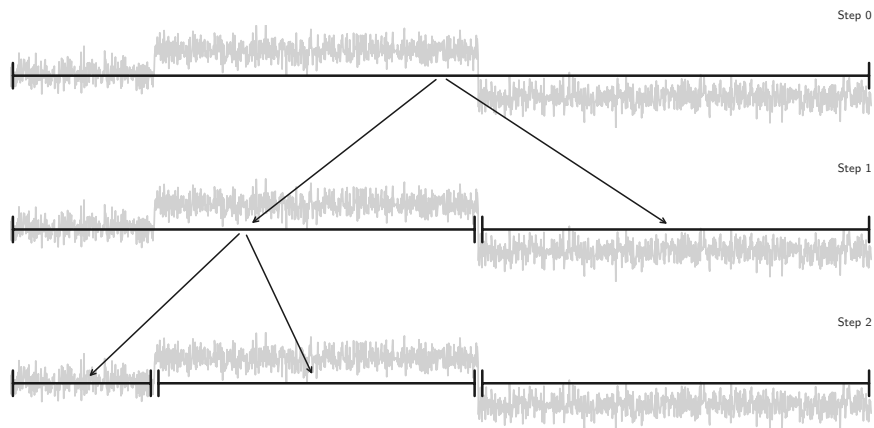


Figure 9: Schematic example of **BinSeg**

- Another extension of **BinSeg** is the wild binary segmentation algorithm [128]. 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, ie those that lower the most the sum of costs. An important parameter of this method is the number of random sub-segments to draw. Wild binary search is combined with c_{L_2} C2 to detect mean-shifts of univariate piecewise constant signals (up to 2000 samples) [128].

5.2.3. Bottom-up segmentation

Bottom-up segmentation, denoted **BotUp**, is the natural counterpart of **BinSeg**. 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 measure $d(\cdot, \cdot)$, defined in 24, between the segments they separate. Change points with the lowest discrepancy are then deleted, meaning that the segments they separate are merged. **BotUp** is often dubbed a "generous" method, by opposition to **BinSeg**, which is "greedy" [134]. A schematic view of the algorithm is displayed on Figure 10 and an implementation is provided in Algorithm 5. Its benefits are its linear computational complexity and conceptual simplicity. However, 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 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 the-

Algorithm 4 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 , ie
 $L = \{t_1, \dots, t_k\}$.
 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
 $\hat{i} \leftarrow \operatorname{argmax}_i G[i]$
 $\hat{t} \leftarrow \operatorname{argmin}_{t_i < t < t_{i+1}} [c(y_{t_i..t}) + c(y_{t..t_{i+1}})]$.
 $L \leftarrow L \cup \{\hat{t}\}$
until stopping criterion is met.
Output: set L of estimated breakpoint indexes.

824 oretical convergence study is available. It has been applied on speech time series
825 to detect mean and scale shifts [120]. Besides, the authors of [134] have found
826 that **BotUp** outperforms **BinSeg** on ten different data sets such as physiologi-
827 cal signals (ECG), financial time-series (exchange rate), industrial monitoring
828 (water levels), etc.

6. Estimating the number of changes

830 This section presents the third defining element of change detection meth-
831 ods, namely the constraint on the number of change points. Here, the number
832 of change points is assumed to be unknown (P2). Existing procedures are or-
833 ganized by the penalty function that they are based on. Common heuristics
834 are also described. The organization of this section is schematically shown in
835 Figure 11.

6.1. Linear penalty

836 Arguably the most popular choice of penalty [113], the linear penalty (also
837 known as l_0 penalty) generalizes several well-known criteria from the literature
838 such as the Bayesian Information Criterion (BIC) and the Akaike Information
839 Criterion (AIC) [135, 136]. The linear penalty, denoted pen_{l_0} , is formally defined
840 as follows.
841

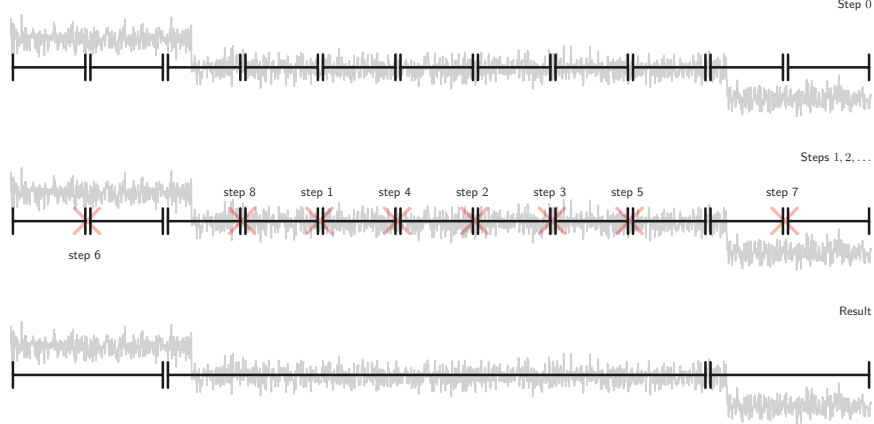


Figure 10: Schematic view of BotUp

842 **Penalty 1 (pen_{l_0}).** The penalty function pen_{l_0} is given by

$$\text{pen}_{l_0}(\mathcal{T}) := \beta|\mathcal{T}| \quad (28)$$

843 where $\beta > 0$ is the smoothing parameter.

844 Intuitively, the smoothing parameter controls the trade-off between complexity
 845 and goodness-of-fit (measured by the sum of costs): low values of β favour
 846 segmentations with many regimes and high values of β discard most change
 847 points.

848 *Calibration..* From a practical standpoint, once the cost function has been cho-
 849 sen, the only parameter to calibrate is the smoothing parameter. Several ap-
 850 proaches, based on model selection, can be found in the literature: they assume
 851 a model on the data, for instance (M1), (M2), (M3), and choose a value of β
 852 that optimizes a certain statistical criterion. The best-known example of such
 853 an approach is BIC, which aims at maximizing the constrained log-likelihood of
 854 the model. The exact formulas of several linear penalties, derived from model
 855 selection procedures, are given the following paragraph. Conversely, when no
 856 model is assumed, different heuristics are applied to tune the smoothing param-
 857 eter. For instance, one can use a procedure based on cross-validation [137] or
 858 the slope heuristics [138]. In [139, 140], supervised algorithms are proposed: the
 859 chosen β is the one that minimizes an approximation of the segmentation error
 860 on an annotated set of signals.

861 *Related penalties..* A number of model selection criteria are special cases of
 862 the linear penalty pen_{l_0} . For instance, under Model (M1) (iid with piecewise
 863 constant distribution), the constrained likelihood that is derived from the BIC
 864 and the penalized sum of costs are formally equivalent, upon setting $c = c_{\text{i.i.d.}}$
 865 and $\text{pen} = \text{pen}_{\text{BIC}}$, where pen_{BIC} is defined as follows.

Algorithm 5 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 , ie
 $L = \{t_1, \dots, t_k\}$.
 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
 $\hat{i} \leftarrow \operatorname{argmin}_i G[i]$
 Remove $t_{\hat{i}+1}$ from L .
until stopping criterion is met.
Output: set L of estimated breakpoint indexes.

866 **Penalty 2 (pen_{BIC}).** *The penalty function pen_{BIC} is given by*

$$\text{pen}_{BIC}(\mathcal{T}) := \frac{p}{2} \log T |\mathcal{T}| \quad (29)$$

867 *where $p \geq 1$ is the dimension of the parameter space in (M1).*

868 In the extensively studied model of an univariate Gaussian signal, with fixed
869 variance σ^2 and piecewise constant mean, the penalty pen_{BIC} becomes pen_{L₂},
870 defined below. Historically, it was one of the first penalties introduced for change
871 point detection [135, 141].

872 **Penalty 3 (pen_{BIC,L₂}).** *The penalty function pen_{BIC,L₂} is given by*

$$\text{pen}_{BIC,L_2}(\mathcal{T}) := \sigma^2 \log T |\mathcal{T}|. \quad (30)$$

873 *where σ is the standard deviation and T is the number of samples.*

874 In the same setting, AIC, which is a generalization of Mallows' C_p [62], also
875 yields a linear penalty, namely pen_{AIC,L₂}, defined as follows.

876 **Penalty 4 (pen_{AIC,L₂}).** *The penalty function pen_{AIC,L₂} is given by*

$$\text{pen}_{AIC,L_2}(\mathcal{T}) := \sigma^2 |\mathcal{T}|. \quad (31)$$

877 *where σ is the standard deviation.*

878 6.2. Fused lasso

879 For the special case where the cost function is c_{L_2} , a faster alternative to
880 pen_{l₀} can be used. To that end, the l_0 penalty is relaxed to a l_1 penalty [18, 48].
881 The resulting penalty function, denoted pen_{l₁}, is defined as follows.

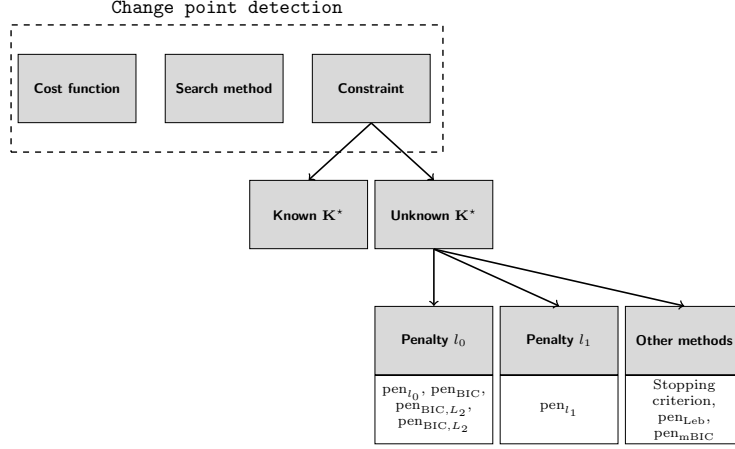


Figure 11: Typology of the constraints (on the number of change points) described in Section 6.

Penalty 5 (pen_{l_1}). The penalty function pen_{l_1} is given by

$$\text{pen}_{l_1}(\mathcal{T}) := \beta \sum_{k=1}^{|\mathcal{T}|} \|\bar{y}_{t_{k-1}..t_k} - \bar{y}_{t_k..t_{k+1}}\|_1 \quad (32)$$

where $\beta > 0$ is the smoothing parameter, the t_k are the elements of \mathcal{T} and $\bar{y}_{t_{k-1}..t_k}$ is the empirical mean of sub-signal $y_{t_{k-1}..t_k}$.

This relaxation strategy (from l_0 to l_1) is shared with many developments in machine learning, for instance sparse regression, compressive sensing, sparse PCA, dictionary learning [84], where pen_{l_1} is also referred to as the fused lasso penalty. In numerical analysis and image denoising, it is also known as the total variation regularizer [13, 18, 48]. Thanks to this relaxation, the optimization of the penalized sum of costs (1) in (P2) is transformed into a convex optimization problem, which can be solved efficiently using Lars (for “least absolute shrinkage and selection operator”) [18, 48]. The resulting complexity is of this order of $\mathcal{O}(T \log T)$ [84, 142]. From a theoretical standpoint, under the mean-shift model (piecewise constant signal with Gaussian white noise), the estimated change point fractions are asymptotically consistent [48]. This result is demonstrated for an appropriately converging sequence of values of β . This consistency property is obtained even though classical assumptions from the Lasso regression framework (such as the irrepresentable condition) are not satisfied [48]. In the literature, pen_{l_1} , combined with c_{L_2} , is applied on DNA sequences [16, 18], speech signals [12] and climatological data [143].

6.3. Complex penalties

Several other penalty functions can be found in the literature. However they are more complex, in the sense that the optimization of the penalized sum of

cost is not tractable. In practice, the solution is found by computing the optimal segmentations with K change points, with $K = 1, 2, \dots, K_{\max}$ for a sufficiently large K_{\max} , and returning the one that minimizes the penalized sum of costs. When possible, the penalty can also be approximated by a linear penalty, in which case, `Pelt` can be used. In this section, we describe two examples of complex penalties. Both originate from theoretical considerations, under the univariate mean-shift model, with the cost function c_{L_2} . The first example is the modified BIC criterion (mBIC) [144], which consists in maximizing the asymptotic posterior probability of the data. The resulting penalty function, denoted pen_{mBIC} , depends on the number and repartition of the change point indexes: intuitively, it favours evenly spaced change points.

Penalty 6 (pen_{mBIC}). *The penalty function pen_{mBIC} is given by*

$$\text{pen}_{\text{mBIC}}(\mathcal{T}) := 3|\mathcal{T}| \log T + \sum_{k=0}^{|\mathcal{T}|+1} \log\left(\frac{t_{k+1} - t_k}{T}\right) \quad (33)$$

where the t_k are the elements of \mathcal{T} .

In [145], a model selection procedure leads to another complex penalty function, namely pen_{Leb} (where “Leb” is short for “Lebarbier”, the author of [145]). Upon using this penalty function, the penalized sum of costs satisfied a so-called oracle inequality, which holds in a non-asymptotic setting, contrary to the other penalties previously described.

Penalty 7 (pen_{Leb}). *The cost function pen_{Leb} is given by*

$$\text{pen}_{\text{Leb}}(\mathcal{T}) := \frac{|\mathcal{T}| + 1}{T} \sigma^2 (a_1 \log \frac{|\mathcal{T}| + 1}{T} + a_2) \quad (34)$$

where $a_1 > 0$ and $a_2 > 0$ are positive parameters and σ^2 is the noise variance.

7. Current challenges in change point detection

Change point detection is an active research field that has adapted to the interests of the scientific community. In particular, to cope with the complexity and the current challenges of real-world data, recent methods have encompassed techniques linked to machine learning and artificial intelligence. In this section, we review two hot topics which have received a lot of attention in the recent years: the change detection for high-dimensional data and the use of supervised learning for the automatic calibration of the detection methods.

7.1. Change detection for high-dimensional signals

In this review, the number of components d was assumed to be fixed and the number T of samples, growing to infinity. However, in certain applications (for instance copy number data, financial time series, internet traffic monitoring), d

936 is of the same order of magnitude as T , or even far larger. This high-dimensional
 937 setting brings about at least two difficulties, first about the consistency of the
 938 breakpoint estimates, and second, about the computational complexity. For
 939 mean-shift detection, total variation regression (cost function c_{L_2} and penalty
 940 function pen_{l_1}) remains a popular choice because it enjoys a low complexity.
 941 In addition, it has been shown to consistently estimate single change points
 942 for the “fixed T increasing d ” setting [18]. Nevertheless, several authors have
 943 observed that for high dimensional signals, certain statistical quantities are no
 944 longer so easily estimated: for instance, the inverse of the sample covariance
 945 matrix is not necessarily an unbiased estimator of the precision matrix, when
 946 d is larger than T (see [146] for example). Since this quantity is often used to
 947 detect mean-shifts in i.i.d. Gaussian signals with arbitrary variance/covariance
 948 matrix Σ (cost function c_M with $M = \Sigma^{-1}$), more robust replacements to similar
 949 conventional statistics were introduced [147, 148]. However, those formulations
 950 overlook the fact that, in the high-dimensional setting, the changes to detect
 951 often occur in only a fraction of the signal components. As argued in [80], this
 952 can lead to underestimation of the number of changes. To that end, procedures
 953 that take this sparsity into account have been proposed. One possible solution
 954 this problem is to search changes in all possible combinations of components,
 955 and aggregate the results in a sensible way. In [149], a procedure for single
 956 mean-shifts is described: the estimation is shown to be consistently for T and d
 957 increasing to infinity, and under certain conditions (in particular, the number of
 958 changing components must be sufficiently large) Also, their method has a linear
 959 complexity (up to a logarithmic factor), despite the combinatorial optimization
 960 problem. In [150], the authors give a general discussion on how to aggregate
 961 (dependant) change detection statistics, each based on a subset of an increasing
 962 number of components. Another method consists in finding a data-driven pro-
 963 jection of the multivariate signal and detect changes on the resulting univariate
 964 signal [151]. This is again a combinatorial problem, which can be relaxed to
 965 make it convex. The change point estimation is shown to be consistent, under
 966 certain conditions. The more general setting of piecewise stationary signals is
 967 also tackled for high dimensional time series. A number of associated proce-
 968 dures, which are meant to detect changes in the second order structure of a
 969 random process, consist in transforming the signal in a sensible way, and then
 970 search mean-shifts with a conventional method [152]. For instance, the trans-
 971 formation can be based on local periodograms [153], or more recently, on PCA
 972 and wavelet transforms [154, 155].

973 *7.2. Supervised approaches for automatic calibration of detection algorithms*

974 The calibration of detection methods is an important and complex step. As
 975 seen in this review, the search method mostly influences the trade-off between
 976 computational complexity and segmentation accuracy, and is relatively easy to
 977 choose. Conversely, the cost function and the constraint are related to (often)
 978 subjective considerations, which are respectively the nature of the changes to
 979 detect, and their amplitudes. Several authors (including the authors of these

lines) have applied supervised learning procedures to find adapted cost functions and constraints, for a given data set and task. Generally speaking, such procedures can infer complex decision rules only using relevant examples. In the segmentation context, this supposes that experts are able to provide such examples: they consist in signals, manually segmented. The objective is to use those examples to design a change point detection method able to replicate the segmentation strategy of the experts. Two types of annotations (or labels, in the supervised learning terminology) are often considered: full and partial. For a fully annotated signal, the timestamps of all changes are provided by the expert. For a partially annotated signal, the expert only gives an approximate localization of the change points (exact locations are not known). This setting was first studied in [16], then extended in [31, 140]. The objective was to find the optimal constraint to detect an unknown number of changes, using a few breakpoint annotations. As for the calibration of the cost function, the works of [30] and [107] propose a metric learning approach.

8. Summary table

This 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. Note that `Pelt` (introduced in 2012) is sometimes associated with publications prior to 2012. It is because some linear penalties [62, 144] were introduced long before `Pelt` was, and authors then resorted to quadratic (at best) algorithms. Nowadays, the same results can be obtained faster with `Pelt`. A guide of computational complexity is also provided. 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.)

Publication	Search method	Cost function	Known K Yes No	Scalability (wrt T)	Package	Additional information
Sen and Srivastava (1975), Vostrikova (1981)	BiasSeg	c_{L_2}	✓ -	★★★	✓	
Yao (1988)	Opt	c_{L_2}	- ✓	★★☆	-	Bayesian information criterion (BIC)
Basseville and Nikiforov (1993)	Opt	$c_{i.i.d.}, c_{L_2}$	-	★★★	-	single change point
Bai (1994), Bai and Perron (2003)	Opt	c_{linear}, L_2	-	★★☆	-	single change point
Bai (1995)	Opt	c_{linear}, L_1	-	★★☆	-	single change point
Lavielle (1998)	Opt	c_{AR}	✓ -	★★☆	-	
Bai (2000)	Opt	c_{AR}	✓ -	★★☆	-	
Birgé and Massart (2001), Birgé and Massart (2007)	Opt	c_{L_2}	- ✓	★★☆	-	model selection
Bai and Perron (2003)	Opt	c_{L_2}	✓ -	★★☆	-	
Olshen et al. (2004), Venkatraman and Olshen (2007)	BiasSeg	c_{L_2}	✓ ✓	★★★	✓	
Lebarbier and Lebarbier (2005)	Opt	c_{L_2}	- ✓	★★☆	-	model selection
Desobry et al. (2005)	Win	c_{kernel}	- ✓	★★★	-	dissimilarity measure (one-class SVM), see Remark 4
Harchaoui and Cappé (2007)	Opt	c_{kernel}, c_{rbf}	✓ -	★★☆	-	
Zhang and Siegmund (2007)	Pelt	c_{L_2}	- ✓	★★☆	-	modified BIC
Harchaoui et al. (2009)	Win	-	✓ ✓	★★★	-	dissimilarity measure (Fisher discriminant), see Remark 4
Lévy-Leduc and Rounoff (2009), Lung-Yut-Fong et al. (2012)	Win	c_{rank}	✓ ✓	★★★	✓	dissimilarity measure (rank-based), see Remark 4
Bai (2010)	Opt	c_{L_2}, c_{Σ}	- ✓	★★☆	-	single change point
Vert and Bleakley (2010)	Fused Lasso	c_{L_2}	- ✓	★★★	-	Tikhonov regularization
Harchaoui and Lévy-Leduc (2010)	Fused Lasso	c_{L_2}	- ✓	★★★	-	total variation regression (pen_{l_1})
Arlot et al. (2012)	Opt	c_{kernel}, c_{rbf}	✓ ✓	★★☆	-	
Killick et al. (2012)	Pelt	any $c(\cdot)$	- ✓	★★☆	✓	
Angelosante and Giannakis (2012)	Fused Lasso	c_{AR}	- ✓	★★★	-	Tikhonov regularization
Liu et al. (2013)	Win	-	- ✓	★★★	-	dissimilarity measure (density ratio), see Remark 4
Hocking et al. (2013)	Pelt	c_{L_2}	- ✓	★★☆	-	supervised method to learn a penalty level (pen_{l_0})
Fryzlewicz (2014)	BiasSeg	c_{L_2}	✓ ✓	★★★	✓	univariate signal
Lajugie et al. (2014)	Opt	c_M	✓ -	★★☆	-	supervised method to learn a suitable metric
Frick et al. (2014)	BiasSeg	$c_{i.i.d.}$	✓ ✓	★★★	✓	exponential distributions family
Lung-Yut-Fong et al. (2015)	Opt	c_{rank}	✓ -	★★☆	✓	
Garreau and Arlot (2018)	Pelt	c_{kernel}, c_{rbf}	✓ ✓	★★☆	-	
Haynes et al. (2017)	Pelt	any $c(\cdot)$	- ✓	★★☆	-	
Chakar et al. (2017)	Pelt	c_{AR}	✓ ✓	★★☆	✓	

Table 2: Summary table of literature review.

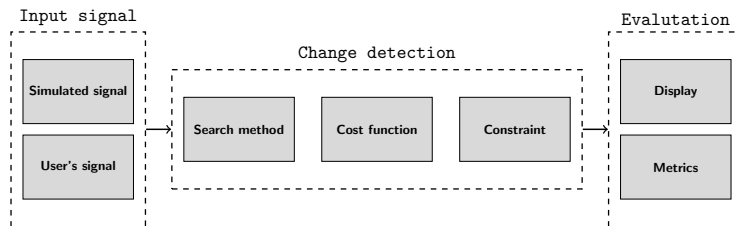


Figure 12: Schematic view of the `ruptures` package.

9. Presentation of the Python package

Most of the approaches presented in this article are included in a Python scientific library for multiple change point detection in multivariate (or univariate depending on the used cost function) signals called `ruptures` [37]. The `ruptures` library is written in pure Python and available on Mac OS X, Linux and Windows platforms. Source code is available from [37] under the BSD license and deployed with a complete documentation that includes installation instructions and explanations with code snippets on advance use.

A schematic view is displayed on Figure 12. Each block of this diagram is described in the following brief overview of `ruptures`' features.

- **Search methods** Our package includes the main algorithms from the literature, namely dynamic programming, detection with a l_0 constraint, binary segmentation, bottom-up segmentation and window-based segmentation. This choice is the result of a trade-off between exhaustiveness and adaptiveness. Rather than providing as many methods as possible, only algorithms which have been used in several different settings are included. In particular, numerous “mean-shift only” detection procedures were not considered. Implemented algorithms have sensible default parameters that can be changed easily through the functions' interface.
- **Cost functions** Cost functions are related to the type of change to detect. Within `ruptures`, one has access to parametric cost functions that can detect shifts in standard statistical quantities (mean, scale, linear relationship between dimensions, autoregressive coefficients, etc.) and non-parametric cost functions (kernel-based or Mahalanobis-type metric) that can, for instance, detect distribution changes [30, 97]. Note that in the case of piecewise linear regression cost functions, only univariate signal may be used as inputs.
- **Constraints** All methods can be used whether the number of change points is known or not. In particular, `ruptures` implements change point detection under a cost budget and with a linear penalty term [17, 113].
- **Evaluation** Evaluation metrics are available to quantitatively compare segmentations, as well as a display module to visually inspect algorithms' performances.

- 1042 • **Input** Change point detection can be performed on any univariate or
1043 multivariate signal that fits into a *Numpy* array. A few standard non-
1044 stationary signal generators are included.
- 1045 • **Consistent interface and modularity** Discrete optimization methods
1046 and cost functions are the two main ingredients of change point detection.
1047 Practically, each is related to a specific object in the code, making the code
1048 highly modular: available optimization methods and cost functions can be
1049 connected and composed. An appreciable by-product of this approach is
1050 that a new contribution, provided its interface follows a few guidelines,
1051 can be integrated seamlessly into **ruptures**.
- 1052 • **Scalability** Data exploration often requires to run several times the same
1053 methods with different sets of parameters. To that end, a cache is imple-
1054 mented to keep intermediate results in memory, so that the computational
1055 cost of running the same algorithm several times on the same signal is
1056 greatly reduced. We also add the possibility for a user with speed con-
1057 straints to sub-sample their signals and set a minimum distance between
1058 change points.

1059 10. Conclusion

1060 In this article, we have reviewed numerous methods to perform change point
1061 detection, organized within a common framework. Precisely, all methods are
1062 described as a collection of three elements: a cost function, a search method
1063 and a constraint on the number of changes to detect. This approach is in-
1064 tended to facilitate prototyping of change point detection methods: for a given
1065 segmentation task, one can pick among the described elements to design an
1066 algorithm that fits its use-case. Most detection procedures described above are
1067 available within the Python language from the package **ruptures** [37], which is
1068 the most comprehensive change point detection library. Its consistent interface
1069 and modularity allow painless comparison between methods and easy integra-
1070 tion of new contributions. In addition, a thorough documentation is available
1071 for novice users. Thanks to the rich Python ecosystem, **ruptures** can be used
1072 in coordination with numerous other scientific libraries .

1073 Acknowledgements

1074 This work was supported by a public grant as part of the Investissement
1075 d’avenir project, reference ANR-11-LABX-0056-LMH, LabEx LMH.

1076 References

- 1077 [1] E. S. Page. Continuous inspection schemes. *Biometrika*, 41:100–105, 1954.
- 1078 [2] E. S. Page. A test for a change in a parameter occurring at an unknown point.
1079 *Biometrika*, 42:523–527, 1955.
- 1080 [3] M. Basseville and I. Nikiforov. *Detection of abrupt changes: theory and application*,
1081 volume 104. Prentice Hall Englewood Cliffs, 1993.
- 1082 [4] B. E. Brodsky and B. S. Darkhovsky. *Nonparametric methods in change point problems*.
1083 Springer Netherlands, 1993.
- 1084 [5] M. Csörgö and L. Horváth. *Limit theorems in change-point analysis*. Chichester, New
1085 York, 1997.
- 1086 [6] Jie Chen and Arjun K. Gupta. *Parametric Statistical Change Point Analysis: With*
1087 *Applications to Genetics, Medicine, and Finance*. Springer Science & Business Media,
1088 2011.
- 1089 [7] M. Lavielle and G. Teyssière. Adaptive detection of multiple change-points in asset
1090 price volatility. In *Long-Memory in Economics*, pages 129–156. Springer Verlag, Berlin,
1091 Germany, 2007.
- 1092 [8] Venkata Jandhyala, Stergios Fotopoulos, Ian Macneill, and Pengyu Liu. Inference for
1093 single and multiple change-points in time series. *Journal of Time Series Analysis*, 34
1094 (4):423–446, 2013. ISSN 01439782. doi: 10.1111/jtsa.12035.
- 1095 [9] K. Haynes, I. A. Eckley, and P. Fearnhead. Computationally efficient changepoint
1096 detection for a range of penalties. *Journal of Computational and Graphical Statistics*,
1097 26(1):134–143, 2017.
- 1098 [10] F. Desobry, M. Davy, and C. Doncarli. An online kernel change detection algorithm.
1099 *IEEE Transactions on Signal Processing*, 53(8):2961–2974, 2005.
- 1100 [11] Z. Harchaoui, F. Vallet, A. Lung-Yut-Fong, and O. Cappé. A regularized kernel-based
1101 approach to unsupervised audio segmentation. In *Proceedings of the IEEE International*
1102 *Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1665–1668,
1103 Taipei, Taiwan, 2009.
- 1104 [12] D. Angelosante and G. B. Giannakis. Group lassoing change-points piece-constant AR
1105 processes. *EURASIP Journal on Advances in Signal Processing*, 70, 2012.
- 1106 [13] N. Seichepine, S. Essid, C. Fevotte, and O. Cappé. Piecewise constant nonnegative
1107 matrix factorization. In *Proceedings of the IEEE International Conference on Acoustics,*
1108 *Speech and Signal Processing (ICASSP)*, pages 6721–6725, Florence, Italy, 2014.
- 1109 [14] J. Bai and P. Perron. Estimating and testing linear models with multiple structural
1110 changes. *Econometrica*, 66(1):47–78, 1998.
- 1111 [15] K. Frick, A. Munk, and H. Sieling. Multiscale change point inference. *Journal of the*
1112 *Royal Statistical Society. Series B: Statistical Methodology*, 76(3):495–580, 2014.
- 1113 [16] T. Hocking, G. Schleiermacher, I. Janoueix-Lerosey, V. Boeva, J. Cappel, O. Delattre,
1114 F. Bach, and J.-P. Vert. Learning smoothing models of copy number profiles using
1115 breakpoint annotations. *BMC Bioinformatics*, 14(1):164, 2013.
- 1116 [17] R. Maidstone, T. Hocking, G. Rigai, and P. Fearnhead. On optimal multiple change-
1117 point algorithms for large data. *Statistics and Computing*, 27(2):519–533, 2017.

- 1118 [18] J.-P. Vert and K. Bleakley. Fast detection of multiple change-points shared by many
1119 signals using group LARS. In *Advances in Neural Information Processing Systems*
1120 (*NIPS*), volume 1, pages 2343–2351, Vancouver, Canada, 2010.
- 1121 [19] F. Picard, S. Robin, M. Lavielle, C. Vaisse, and J.-J. Daudin. A statistical approach for
1122 array CGH data analysis. *BMC Bioinformatics*, 6(1):27, 2005.
- 1123 [20] Y. Guédon. Exploring the latent segmentation space for the assessment of multiple
1124 change-point models. *Computational Statistics*, 28(6):2641–2678, 2013.
- 1125 [21] S. Chakar, É. Lebarbier, C. Levy-Leduc, and S. Robin. A robust approach for estimating
1126 change-points in the mean of an AR(1) process. *Bernoulli Society for Mathematical*
1127 *Statistics and Probability*, 23(2):1408–1447, 2017.
- 1128 [22] L. Oudre, R. Barrois-Müller, T. Moreau, C. Truong, R. Dadashi, T. Grégory, D. Ri-
1129 card, N. Vayatis, C. De Waele, A. Yelnik, and P.-P. Vidal. Détection automatique des
1130 pas à partir de capteurs inertiels pour la quantification de la marche en consultation.
1131 *Neurophysiologie Clinique/Clinical Neurophysiology*, 45(4-5):394, 2015.
- 1132 [23] J. Audiffren, R. Barrois-Müller, C. Provost, É. Chiarovano, L. Oudre, T. Moreau,
1133 C. Truong, A. Yelnik, N. Vayatis, P.-P. Vidal, C. De Waele, S. Buffat, and D. Ri-
1134 card. Évaluation de l’équilibre et prédiction des risques de chutes en utilisant une Wii
1135 board balance. *Neurophysiologie Clinique/Clinical Neurophysiology*, 45(4-5):403, 2015.
- 1136 [24] S. Liu, A. Wright, and M. Hauskrecht. Change-point detection method for clinical
1137 decision support system rule monitoring. *Artificial Intelligence In Medicine*, 91:49–56,
1138 2018.
- 1139 [25] R. Maidstone. *Efficient analysis of complex changepoint problems*. PhD thesis, Lan-
1140 caster University, 2016.
- 1141 [26] Jan Verbesselt, Rob Hyndman, Glenn Newnham, and Darius Culvenor. Detecting trend
1142 and seasonal changes in satellite images time series. *Remote Sensing of Environment*,
1143 (114):106–115, 2010.
- 1144 [27] J. Reeves, J. Chen, X. L. Wang, R. Lund, and Q. Q. Lu. A review and comparison of
1145 changepoint detection techniques for climate data. *Journal of Applied Meteorology and*
1146 *Climatology*, 46(6):900–915, 2007. ISSN 15588424. doi: 10.1175/JAM2493.1.
- 1147 [28] C. Lévy-Leduc and F. Roueff. Detection and localization of change-points in high-
1148 dimensional network traffic data. *The Annals of Applied Statistics*, 3(2):637–662, 2009.
- 1149 [29] A. Lung-Yut-Fong, C. Lévy-Leduc, and O. Cappé. Distributed detection/localization of
1150 change-points in high-dimensional network traffic data. *Statistics and Computing*, 22
1151 (2):485–496, 2012.
- 1152 [30] R. Lajugie, F. Bach, and S. Arlot. Large-margin metric learning for constrained par-
1153 titioning problems. In *Proceedings of the 31st International Conference on Machine*
1154 *Learning (ICML)*, pages 297–395, Beijing, China, 2014.
- 1155 [31] T. Hocking, G. Rigail, and G. Bourque. PeakSeg: constrained optimal segmentation
1156 and supervised penalty learning for peak detection in count data. In *Proceedings of the*
1157 *International Conference on Machine Learning (ICML)*, pages 324–332, Lille, France,
1158 2015.
- 1159 [32] R. Barrois-Müller, D. Ricard, L. Oudre, L. Tlili, C. Provost, A. Vienne, P.-P. Vidal,
1160 S. Buffat, and A. Yelnik. Étude observationnelle du demi-tour à l’aide de capteurs iner-
1161 tiels chez les sujets victimes d’AVC et relation avec le risque de chute. *Neurophysiologie*
1162 *Clinique/Clinical Neurophysiology*, 46(4):244, 2016.

- [33] R. Barrois-Müller, T. Gregory, L. Oudre, T. Moreau, C. Truong, A. Aram Pulini, A. Vienne, C. Labourdette, N. Vayatis, S. Buffat, A. Yelnik, C. de Waele, S. Laporte, P.-P. Vidal, and D. Ricard. An automated recording method in clinical consultation to rate the limp in lower limb osteoarthritis. *PLoS One*, 11(10):e0164975, 2016.
- [34] L. Oudre, R. Barrois-Müller, T. Moreau, C. Truong, A. Vienne-Jumeau, D. Ricard, N. Vayatis, and P.-P. Vidal. Template-based step detection with inertial measurement units. *Sensors*, 18(11), 2018.
- [35] C. Truong, L. Oudre, and N. Vayatis. Segmentation de signaux physiologiques par optimisation globale. In *Proceedings of the Groupe de Recherche et d’Etudes en Traitement du Signal et des Images (GRETSI)*, Lyon, France, 2015.
- [36] R. Barrois-Müller, L. Oudre, T. Moreau, C. Truong, N. Vayatis, S. Buffat, A. Yelnik, C. de Waele, T. Gregory, S. Laporte, P. P. Vidal, and D. Ricard. Quantify osteoarthritis gait at the doctor’s office: a simple pelvis accelerometer based method independent from footwear and aging. *Computer Methods in Biomechanics and Biomedical Engineering*, 18 Suppl 1:1880–1881, 2015.
- [37] Charles Truong. ruptures: change point detection in python, 2018. URL <http://ctruong.perso.math.cnrs.fr/ruptures>. [Online].
- [38] L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.
- [39] 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.
- [40] A. F. Martínez and R. H. Mena. On a Nonparametric Change Point Detection Model in Markovian Regimes. *Bayesian Analysis*, 9(4):823–858, 2014.
- [41] D. Barry and J. A. Hartigan. Product partition models for change point problems. *The Annals of Statistics*, 20(1):260–279, 1992.
- [42] D. Barry and J. A. Hartigan. A bayesian analysis for change point problems. *Journal of the American Statistical Association*, 88(421):309–319, 1993.
- [43] S. Aminikhanghahi and D. J. Cook. A survey of methods for time series change point detection. *Knowledge and information systems*, 51(2):339–367, 2017.
- [44] Y. S. Niu, N. Hao, and H. Zhang. Multiple change-point detection: a selective overview. *Statistica Sciences*, 31(4):611–623, 2016.
- [45] J. Bai and P. Perron. Multiple structural change models: a simulation analysis. *Journal of Applied Econometrics*, 18:1–22, 2003.
- [46] 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>.
- [47] 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.
- [48] 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.
- [49] M. Lavielle. Detection of multiples changes in a sequence of dependant variables. *Stochastic Processes and their Applications*, 83(1):79–102, 1999.

- [50] F. Pein, H. Sieling, and A. Munk. Heterogeneous change point inference. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 79(4):1207–1227, 2017.
- [51] H. Keshavarz, C. Scott, and X. Nguyen. Optimal change point detection in Gaussian processes. *Journal of Statistical Planning and Inference*, 193:151–178, 2018.
- [52] 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.
- [53] A. Sen and M. S. Srivastava. On tests for detecting change in mean. *The Annals of Statistics*, 3(1):98–108, 1975.
- [54] P. R. Krishnaiah. Review about estimation of change points. *Handbook of Statistics*, 7:375–402, 1988.
- [55] A. Aue and L. Horváth. Structural breaks in time series. *Journal of Time Series Analysis*, 34:1–16, 2012.
- [56] P Fearnhead. Exact and efficient Bayesian inference for multiple changepoint problems. *Statistics and Computing*, 16(2):203–213, 2006.
- [57] T. Górecki, L. Horváth, and P. Kokoszka. Change point detection in heteroscedastic time series. *Econometrics and Statistics*, 7:63–88, 2018.
- [58] Y.-X. Fu and R. N. Curnow. Maximum likelihood estimation of multiple change points. *Biometrika*, 77(3):563–573, 1990.
- [59] H. He and T. S. Severini. Asymptotic properties of maximum likelihood estimators in models with multiple change points. *Bernoulli*, 16(3):759–779, 2010.
- [60] 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.
- [61] G. Lorden. Procedures for reacting to a change in distribution. *The Annals of Mathematical Statistics*, 42(6):1897–1908, 1971.
- [62] C. L. Mallows. Some comments on Cp. *Technometrics*, 15(4):661–675, 1973.
- [63] G. Hébrail, B. Hugueney, Y. Lechevallier, and F. Rossi. Exploratory analysis of functional data via clustering and optimal segmentation. *Neurocomputing*, 73(7-9):1125–1141, 2010.
- [64] S. Chib. Estimation and comparison of multiple change-point models. *Journal of Econometrics*, 86(2):221–241, 1998.
- [65] M. Lavielle. Using penalized contrasts for the change-point problem. *Signal Processing*, 85(8):1501–1510, 2005.
- [66] J. Bai. Least squares estimation of a shift in linear processes. *Journal of Time Series Analysis*, 15(5):453–472, 1994.
- [67] J. Bai. Least absolute deviation of a shift. *Econometric Theory*, 11(3):403–436, 1995.
- [68] J. Bai. Testing for parameter constancy in linear regressions: an empirical distribution function approach. *Econometrica*, 64(3):597–622, 1996.
- [69] J. Bai. Vector autoregressive models with structural changes in regression coefficients and in variancecovariance matrices. *Annals of Economics and Finance*, 1(2):301–336, 2000.

- [70] J. Bai. Estimation of a change-point in multiple regression models. *Review of Economic and Statistics*, 79(4):551–563, 1997.
- [71] Z. Qu and P. Perron. Estimating and testing structural changes in multivariate regressions. *Econometrica*, 75(2):459–502, 2007.
- [72] Jushan Junshan Bai and Pierre Perron. Computation and analysis of multiple structural change models. *Journal of applied econometrics*, 18(1):1–22, 2003.
- [73] J. Bai. Estimation of multiple-regime regressions with least absolute deviation. *Journal of Statistical Planning and Inference*, 74:103–134, 1998.
- [74] J. Bai. Likelihood ratio tests for multiple structural changes. *Journal of Econometrics*, 91(2):299–323, 1999.
- [75] J. Bai and P. Perron. Critical values for multiple structural change tests. *Econometrics Journal*, 6(1):72–78, 2003.
- [76] P. Perron. Dealing with structural breaks. *Palgrave handbook of econometrics*, 1(2):278–352, 2006.
- [77] J. Bai. Common breaks in means and variances for panel data. *Journal of Econometrics*, 157:78–92, 2010.
- [78] J. Bai, R. L. Lumsdaine, and J. H. Stock. Testing for and dating common breaks in multivariate time series. *The Review of Economic Studies*, 65(3):395–432, 1998.
- [79] P. Perron and Z. Qu. Estimating restricted structural change models. *Journal of Econometrics*, 134(2):373–399, 2006.
- [80] C. Han and A. Taamouti. Partial structural break identification. *Oxford Bulletin of Economics and Statistics*, 79(2):145–164, 2017.
- [81] 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.
- [82] 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.
- [83] P. C. Mahalanobis. On the generalised distance in statistics. *Proceedings of the National Institute of Sciences of India*, 2(1):49–55, 1936.
- [84] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*. Springer, 2nd edition, 2009.
- [85] 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 (NIPS)*, pages 521–528, 2003.
- [86] 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, Corvallis, Oregon, USA, 2007.
- [87] J. H. J. Einmahl and I. W. McKeague. Empirical likelihood based hypothesis testing. *Bernoulli*, 9(2):267–290, 2003.
- [88] C. Zou, G. Yin, F. Long, and Z. Wang. Nonparametric maximum likelihood approach to multiple change-point problems. *The Annals of Statistics*, 42(3):970–1002, 2014.
- [89] K. Haynes, P. Fearnhead, and I. A. Eckley. A computationally efficient nonparametric approach for changepoint detection. *Statistics and Computing*, 27:1293–1305, 2017.

- [90] J. Zhang. Powerful two-sample tests based on the likelihood ratio. *Technometrics*, 48(1):95–103, 2006.
- [91] S. Clemencon, M. Depecker, and N. Vayatis. AUC optimization and the two-sample problem. In *Advances in Neural Information Processing Systems (NIPS)*, pages 360–368, Vancouver, Canada, 2009.
- [92] J. H. Friedman and L. C. Rafsky. Multivariate Generalizations of Wald-Wolfowitz and Smirnov two-sample tests. *The Annals of Statistics*, 7(4):697–717, 1979.
- [93] A. Lung-Yut-Fong, C. Lévy-Leduc, and O. Cappé. Homogeneity and change-point detection tests for multivariate data using rank statistics. *Journal de la Société Française de Statistique*, 156(4):133–162, 2015.
- [94] F. Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945.
- [95] E. L. Lehman and J. P. Romano. *Testing Statistical Hypotheses*, volume 101. springer, 3 edition, 2006.
- [96] M. G. Kendall. *Rank correlation methods*. Charles Griffin, London, England, 1970.
- [97] Z. Harchaoui and O. Cappé. Retrospective multiple change-point estimation with kernels. In *Proceedings of the IEEE/SP Workshop on Statistical Signal Processing*, pages 768–772, Madison, Wisconsin, USA, 2007.
- [98] B. Schölkopf and A. Smola. *Learning with kernels*. MIT Press, Cambridge, USA, 2002.
- [99] 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.
- [100] A. Celisse, G. Marot, M. Pierre-Jean, and G. Rigai. New efficient algorithms for multiple change-point detection with reproducing kernels. *Computational Statistics and Data Analysis*, 128:200–220, 2018.
- [101] 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.
- [102] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge university press, 2004.
- [103] D. Garreau and S. Arlot. Consistent change-point detection with kernels. *Electronic Journal of Statistics*, 12(2):4440–4486, 2018.
- [104] S. Arlot, A. Celisse, and Z. Harchaoui. Kernel change-point detection. *arXiv preprint arXiv:1202.3878*, pages 1–26, 2012.
- [105] J. Cabrieto, F. Tuerlinckx, P. Kuppens, F. H. Wilhelm, M. Liedlgruber, and E. Ceulemans. Capturing correlation changes by applying kernel change point detection on the running correlations. *Information Sciences*, 447:117–139, 2018.
- [106] P. Jain, B. Kulis, J. V. Davis, and I. S. Dhillon. Metric and kernel learning using a linear transformation. *Journal of Machine Learning Research (JMLR)*, 13:519–547, 2012.
- [107] C. Truong, L. Oudre, and N. Vayatis. Supervised kernel change point detection with partial annotations. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, Brighton, UK, 2019.
- [108] S. M. Kay and A. V. Oppenheim. *Fundamentals of Statistical Signal Processing, Volume II: Detection Theory*. Prentice Hall, 1993.

- [109] R. Bellman. On a routing problem. *Quarterly of Applied Mathematics*, 16(1):87–90, 1955.
- [110] M. Lavielle. Optimal segmentation of random processes. *IEEE Transactions on Signal Processing*, 46(5):1365–1373, 1998.
- [111] G. Rigai. 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.
- [112] B. Huguency, G. Hébrail, Y. Lechevallier, and F. Rossi. Simultaneous clustering and segmentation for functional data. In *Proceedings of 16th European Symposium on Artificial Neural Networks (ESANN)*, pages 281–286, Bruges, Belgium, 2009.
- [113] 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.
- [114] 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.
- [115] H. Vullings, M. Verhaegen, and H. Verbruggen. ECG segmentation using time-warping. In *Lecture notes in computer science*, pages 275–286. Springer, 1997.
- [116] 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.
- [117] R. Esteller, G. Vachtsevanos, J. Echaz, 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.
- [118] 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.
- [119] Sudeshna Adak. Time-dependent spectral analysis of nonstationary time series. *Journal of the American Statistical Association*, 93(444):1488–1501, 1998.
- [120] 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.
- [121] 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.
- [122] Z. Harchaoui, F. Bach, and É. Moulines. Kernel change-point analysis. In *Advances in Neural Information Processing Systems (NIPS)*, pages 609–616, Vancouver, Canada, 2008.
- [123] S. Liu, M. Yamada, N. Collier, and M. Sugiyama. Change-point detection in time-series data by relative density-ratio estimation. *Neural Networks*, 43:72–83, 2013.
- [124] 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.
- [125] R. Prescott Adams and D. J. C. MacKay. Bayesian Online Changepoint Detection. Technical report, 2007.

- [126] 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.
- [127] J. Bai. Estimating multiple breaks one at a time. *Econometric Theory*, 13(3):315–352, 1997.
- [128] Piotr Fryzlewicz. Wild binary segmentation for multiple change-point detection. *Annals of Statistics*, 42(6):2243–2281, 2014. ISSN 00905364. doi: 10.1214/14-AOS1245.
- [129] 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.
- [130] Y. S. Niu and H. Zhang. The screening and ranking algorithm to detect DNA copy number variations. *The Annals of Applied Statistics*, 6(3):1306–1326, 2012.
- [131] 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.
- [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. 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.
- [134] 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, San Jose, California, USA, 2001.
- [135] Y.-C. Yao. Estimating the number of change-points via Schwarz’ criterion. *Statistics and Probability Letters*, 6(3):181–189, 1988.
- [136] Y.-C. Yao and S. T. Au. Least-squares estimation of a step function. *Sankhy: The Indian Journal of Statistics, Series A*, 51(3):370–381, 1989.
- [137] S. Arlot and A. Celisse. A survey of cross-validation procedures for model selection. *Statistical Surveys*, 4:40–79, 2010.
- [138] L. Birgé and P. Massart. Minimal penalties for Gaussian model selection. *Probability Theory and Related Fields*, 138(1):33–73, 2007.
- [139] T. Hocking, G. Rigai, 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.
- [140] C. Truong, L. Oudre, and N. Vayatis. Penalty learning for changepoint detection. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, pages 1569–1573, Kos, Greece, 2017.
- [141] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978.
- [142] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996.
- [143] J.-J. Jeon, J. Hyun Sung, and E.-S. Chung. Abrupt change point detection of annual maximum precipitation using fused lasso. *Journal of Hydrology*, 538:831–841, 2016.

- [144] 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.
- [145] É. Lebarbier and E. Lebarbier. Detecting multiple change-points in the mean of gaussian process by model selection. *Signal Processing*, 85(4):717–736, 2005. ISSN 01651684. doi: 10.1016/j.sigpro.2004.11.012.
- [146] J. Bai and S. Shi. Estimating high dimensional covariance matrices and its applications. *Annals of Economics and Finance*, 12(2):199–215, 2011. ISSN 15297373.
- [147] Z. Bai and H. Saranadasa. Effect of high dimension: by an example of a two sample problem. *Statistica Sinica*, 6(2):311–329, 1996.
- [148] S. X. Chen and Y. L. Qin. A two-sample test for high-dimensional data with applications to gene-set testing. *The Annals of Statistics*, 38(2):808–835, 2010.
- [149] F. Enikeeva and Z. Harchaoui. High-dimensional change-point detection with sparse alternatives. *arXiv preprint arXiv:1312.1900*, pages 1–33, 2014.
- [150] M. Jirak. Uniform change point tests in high dimension. *The Annals of Statistics*, 43(6):2451–2483, 2015.
- [151] T. Wang and R. J. Samworth. High dimensional change point estimation via sparse projection. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 80(1):57–83, 2018.
- [152] M. Jirak. Change-point analysis in increasing dimension. *Journal of Multivariate Analysis*, 111:136–159, 2012.
- [153] H. Cho and P. Fryzlewicz. Multiple change-point detection for high dimensional time series via sparsified binary segmentation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 77(2):475–507, 2014.
- [154] M. Barigozzi, H. Cho, and P. Fryzlewicz. Simultaneous multiple change-point and factor analysis for high-dimensional time series. *Journal of Econometrics*, 206(1):187–225, 2018.
- [155] S. Ma and L. Su. Estimation of large dimensional factor models with an unknown number of breaks. *Journal of Econometrics*, 207(1):1–29, 2018.
- [156] L. Y. Vostrikova. Detecting disorder in multidimensional random processes. *Soviet Math. Dokl.*, 24:55–59, 1981.
- [157] L. Birgé and P. Massart. Gaussian model selection. *Journal of the European Mathematical Society*, 3(3):203–268, 2001.