Bayesian Model Selection for Change Point Detection and Clustering

Othmane Mazhar [1] **Cristian R. Rojas** [1] **Carlo Fischione** [1] **Mohammad Reza Hesamzadeh** [1]

Abstract

We address a generalization of change point detection with the purpose of detecting the change locations and the levels of clusters of a piecewise constant signal. Our approach is to model it as a nonparametric penalized least square model selection on a family of models indexed over the collection of partitions of the design points and propose a computationally efficient algorithm to approximately solve it. Statistically, minimizing such a penalized criterion yields an approximation to the maximum a-posteriori probability (MAP) estimator. The criterion is then analyzed and an oracle inequality is derived using a Gaussian concentration inequality. The oracle inequality is used to derive on one hand conditions for consistency and on the other hand an adaptive upper bound on the expected square risk of the estimator, which statistically motivates our approximation. Finally, we apply our algorithm to simulated data to experimentally validate the statistical guarantees and illustrate its behavior.

1. Introduction

A classical estimation problem in many scientific inquiries is the well-studied change point detection problem where one tries to estimate when some properties of a sequence of random variables changes. This local property is of prime importance in many learning tasks such as signal segmentation (Abou-Elailah et al., 2016; Kim et al., 2009), change point detection in comparative genomics for early cancer diagnosis (Lai et al., 2005), and modeling and forecasting of changes in financial data (Lavielle & TeyssiÃĺre, 2006; Spokoiny, 2009).

For other applications, one needs more than this local answer and is interested in a more general overview of the time series where for instance earlier data samples behave like new ones creating a clustering effect. Examples of this are found in: electricity market data, where prices might have different behavior corresponding to different price regimes that might reappear depending on some triggering events; signal partitioning with some parts of the signal sharing similar properties; and speech segmentation with different alternating sources. Generally speaking, it is of interest in these situations to determine not only the changes but also the clusters for a more precise description of the inhomogeneous time series.

Parametric models for solving the change point detection problem have been proposed in Cleynen & Lebarbier (2014) and Rigaill et al. (2012). However, in dealing with the change point and clustering problem we would naturally require that our solution does not assume any knowledge of the number of changes nor the actual number of clusters, as these numbers would evolve over time, so we expect new changes in the process to happen and new clusters to form as N, the number of samples, grows. Thus, any practical procedure should be able to estimate these numbers and also have adaptive guarantees with respect to how fast these numbers grow. Similar setups for change point detection have been the subject of study by Harchaoui & Cappé (2007), Arlot et al. (2016) and Garreau & Arlot (2017) who use characteristic kernels for detecting changes in the distribution, while from a computational standpoint a more effective implementation has been proposed by Celisse et al. (2017). In this study, we will restrict ourselves to an *iid* (independent and identically distributed) Gaussian sequence model of the data with known variance, noting that the same study can be done using kernels and that the algorithm we develop can be effectively implemented using the same procedure as in (Celisse et al., 2017), as explained later in the paper.

Two other related lines of research, but which we do not explore here, are on-line algorithms for segmentation and L_1-regularized segmentation. We refer the reader to (Tartakovsky et al., 2014) for an extensive review of on-line algorithms. Data segmentation using the L_1-penalty was introduced by Rudin et al. (1992). The one-dimensional case, corresponding to the Fussed LASSO, has been studied in (Tibshirani et al., 2005) and (Rennie & Dobson, 1969) and an efficient algorithm has been proposed by Arnold & Tibshirani (2016). More recent results can be found in (Dalalyan et al., 2017) for the one-dimensional case and (Hütter & Rigollet, 2016) for two-dimensional case.

Main contribution: The generalized setting of change point detection while clustering the segments for sequences

[1]KTH Royal Institute of Technology, Stockholm, Sweden. Correspondence to: Othmane Mazhar <othmane@kth.se>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.

of data points does not seem to have been previously studied. In this work, we propose a two-pass dynamic programming algorithm for selecting an adequate model from a collection of candidate models. We motivate the choice of the algorithm computationally by showing that it runs in $\mathcal{O}(N^2D + D^4)$ time (where D is an upper bound on the number of change points), statistically by showing that it can be seen as an approximation of a computationally hard MAP optimization problem for which we can derive an oracle inequality that guarantees low sample complexity, consistency and adaptivity, and practically by testing the model on simulation data.

Structure of the paper: In Section 2 we formulate the problem as one of nonparametric model selection from a family of models over all partitions of the data set. After some preliminaries and notations are given in Section 3, we propose in Section 4 a two-pass dynamic programming algorithm as a computationally effective relaxation of the optimization criterion and analyze its computational cost. We then put the model selection problem in a Bayesian framework in Section 5, and use a Laplace-type approximation to derive as optimization criterion the maximum a-posteriori probability. In Section 6 we derive an oracle inequality for the criterion that our algorithm is approximating, and study its properties. Experimental results showing that the clusters and segments can be effectively estimated are presented in Section 7 using simulation data.

2. Problem formulation

Let $\mathcal{Y}$ be a measurable space and $Y_1, Y_2, \ldots, Y_N \in \mathcal{Y}$ denote random variables with distributions P_{Y_i}. Our goal is on one hand to detect changes in the sequence of distribution measures $(P_{Y_i})_{i=1}^N$ and on the other hand to cluster the data points coming from the same process. Hence we put random variables between two consecutive changes in the same segment, and we think of random variables of the same segment or different segments as belonging to the same cluster if they are the realization of the same process.

One important case both in theory and in practice is the uniform constant design model were the Y_is depend on deterministic variables uniformly spaced on a grid $X_i = i$ for $i \in [\![1, N]\!] := \{1, \ldots, N\}$ through a regression function f^* with an additive *iid* random noise $(\epsilon_i)_{i=1}^N$. Taking the distribution of the ϵ_i's as $\mathcal{N}(0, \sigma^2)$ with known variance, we end up with the following Gaussian sequence model:

$$Y_i = f_i^* + \epsilon_i, \quad \text{for } i \in [\![1, N]\!]. \tag{1}$$

Here we are placed in a regression setting of the form $Y = f^* + \epsilon$, where $Y = [Y_1 \ \cdots \ Y_N]^T$, $f^* = [f_1^* \ \cdots \ f_N^*]^T$ and $\epsilon = [\epsilon_1 \ \cdots \ \epsilon_N]^T \sim \mathcal{N}(0, \sigma^2 I_N)$, and we are interested in estimating f^* as a piecewise constant function that takes limited number of values.

We emphasize that it is unlikely that the data correspond exactly to a piecewise constant function plus independent random Gaussian noise and that we are in this low dimensional hidden structure exactly, yet there might exist a good sparse linear approximation. Hence our search is not for an exact model, rather we are trying to select the best model in a collection of candidates, as we explain in the next section.

3. Preliminaries and notation

We would like to perform dimensionality reduction by exploiting the hidden structure on the data sequence $Y_1, Y_2, \ldots, Y_N$. To do this we split it into different segments while also putting the segments sharing the same mean into the same cluster. Hence if we knew the clusters our problem reduces to fitting a constant to a set of observations over each cluster. Observe that if f^* is constant over parts of $[\![1, N]\!]$, then it determines a clustering of $Y_1, Y_2, \ldots, Y_N$ over the values where it is constant. Hence, we can think about the problem as, first determining the clustering of the $Y_1, Y_2, \ldots, Y_N$ which would result in a partition π of $[\![1, N]\!]$, and then choosing the best value of $\hat{f}$ over each part as our estimate. So f^* belong to the subspace $\mathcal{F}_\pi$: subspace of functions that are constant over the parts of the partition π.

To formalize this, let $\mathcal{M}$ be an index set over the collection of partitions Π_N of $[\![1, N]\!]$; given $m \in \mathcal{M}$, denote by $\mathcal{F}_m$ the subspace of functions that are constant over the parts of π_m. Our goal is two-fold: find $\hat{m}$ as the index estimate of $\mathcal{F}_{\hat{m}}$, the subspace where the estimate of f^* lives, and from $\mathcal{F}_{\hat{m}}$ compute $\hat{f}_{\hat{m}}$ as our estimate. We represent a partition π as an unordered collection of its subsets $\pi = \{[1], [2], \ldots, [|\pi|]\}$ with $[k]$ being the k^{th}-equivalent class, -part or -cluster, and $|\pi|$ the cardinality of the partition. Every part $[k]$ can be seen as the union of segments $[k] = \{[k_1], [k_2], \ldots, [k_{|[k]|}]\}$ where $(k_i)_{i=1}^{|[k]|}$ is the collection of maximal intervals in $[k]$ that we call segments of the k^{th}-cluster. The last element in each segment $[k_i]$ is called a change point. We define $d'_m := |\pi_m| - 1 = \dim(\mathcal{F}_m) - 1$ as the clustering dimension. Even though this choice might create some confusion it will be consistent the notations used in the proofs of sections 5 and 6. Also we define $d''_m = |\pi_m|_0 := |\cup_{k=1}^{d'_m+1}[k]|$ as the change point dimension.

To link partitions to subspaces let $e_l := (0, \ldots, 1, \ldots, 0)$ be the l^{th}-component of the standard orthonormal basis of $\mathbb{R}^N$, and define for a subset A of $[\![1, N]\!]$ the vector $\mathbb{1}_A := \sum_{l \in A} e_l$. For $[k]$, the k^{th} cluster of π_m, with a slight abuse of notation we define $\mathbb{1}_{[k]} := \sum_{i=1}^{|[k]|} e_{k_i}$, and observe that $\mathcal{F}_m = \operatorname{span}\{\mathbb{1}_{[k]} \colon k \in \pi_m\}$, which is consistent with the definition of the clustering dimension $d'_m := |\pi_m| - 1 = \dim(\mathcal{F}_m) - 1$.

We define $\langle f^* \rangle := \operatorname{span}\{f^*\}$, $\mathcal{S}_1 \oplus \mathcal{S}_2$ as the direct sum of the two vector space $\mathcal{S}_1$ and $\mathcal{S}_2$, and $\mathcal{S}_1 \ominus \mathcal{S}_2$ as their direct difference. $\mathbb{P}_\mathcal{S}$ denotes the (orthogonal) projection operator onto the subspace $\mathcal{S}$. We also define the partitions inclusion as $m_1 \subset m_2$ if $\mathcal{F}_{m_1} \subset \mathcal{F}_{m_2}$, or equivalently if

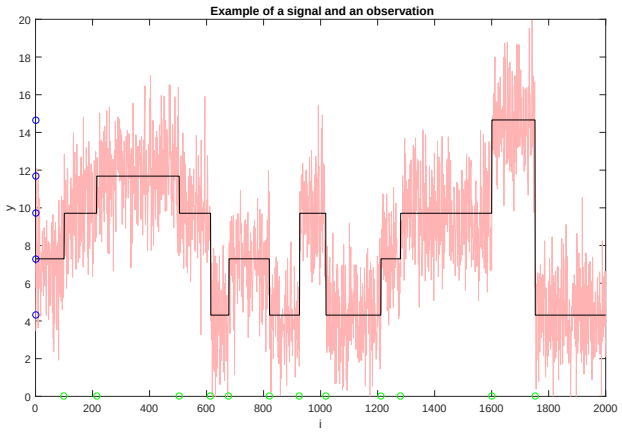

Figure 1. Example of a piecewise constant signal f^* (black line) and observed signal Y (pink line) with clustering values (blue circles) and change points (green circles).

π_{m_2} is finer than π_{m_1}.

Example 1. *Consider the signal f^* of Figure 1, whose partition is*

$$\pi = \{[1]; [2]; [3]; [4]; [5]\},$$

where

$$[1] = [\![615, 678]\!] \cup [\![821, 926]\!] \cup [\![1019, 1211]\!] \cup [\![1753, 2000]\!]$$
$$[2] = [\![1, 100]\!] \cup [\![679, 820]\!] \cup [\![1212, 1280]\!]$$
$$[3] = [\![101, 214]\!] \cup [\![505, 614]\!] \cup [\![926, 1018]\!] \cup [\![1281, 1600]\!]$$
$$[4] = [\![215, 504]\!]$$
$$[5] = [\![1601, 1752]\!].$$

Hence, $d'_\pi = 4$ and $d''_\pi = 12$ for this signal.

We also denote by C_k^N the binomial coefficient that gives the number of ways, disregarding order, that k objects can be chosen from among N objects. This is given by

$$C_k^N := \frac{N!}{k!(N-k)!}. \tag{2}$$

The Stirling numbers of the second kind, $S(N, k)$, correspond to the number of ways to partition a set of N objects into k non-empty subsets, or, similarly, to the number of different equivalence relations with precisely k equivalence classes that can be defined on an set of N elements.

We are precisely interested in the case where the element set is $[\![1, N]\!]$ and the distance between every two elements in each equivalence class is at least 2; we denote the number of such equivalent classes by $S^2(N, k)$. $S(N, k)$ and $S^2(N, k)$ satisfy the following recurrence relations:

$$\begin{aligned} S(N,k) &= S(N-1,k-1) + kS(N-1,k), \quad N \geqslant k, \\ S^2(N,k) &= S(N-1,k-1), \quad N, k \geqslant 2. \end{aligned} \tag{3}$$

For the proofs of these results, we refer the reader to (Graham et al., 1988) and (Mohr & Porter, 2009).

4. Two-pass dynamic programming for change point detection and clustering

To solve the change point and clustering problem, a natural approach is to consider the minimization of a criterion of the form,

$$\mathrm{Crit}(m) = \|y - \hat{f}_m\|_2^2 + \sigma^2 K \,\mathrm{pen}(m). \tag{4}$$

Uniqueness, continuity and stability properties of similar criterion have been studied in (O. et al.), we restrict to a penalty term $\mathrm{pen}(d'_m, d''_m) := \mathrm{pen}(m)$ depending only on d' and d'' and a multiplicative tuning parameter K. Indeed, as we shall see the penalty can be chosen such that the minimizer $\hat{f}_{\hat{m}}$ of (4) behaves like an approximation to a maximum a-posteriori estimator (MAP), and also, the average expected risk $\frac{1}{N}\mathbb{E}[\|\hat{f}_{\hat{m}} - f^*\|_2^2] \to 0$ for a large class of signals f^*, namely, those corresponding to models with $d' \leqslant d'' = o(N/\ln N)$, *i.e.*, f^* is a consistent estimator for those signals. The specific form of $\mathrm{pen}(m)$ will be derived in the next section, based on an oracle inequality that will guarantee consistency and adaptivity of our estimator.

Although the estimator $\hat{f}_{\hat{m}}$ enjoys good statistical properties, from a computational stand it would involve the exploration of $\mathcal{M}$. The set $\mathcal{M}$ is identified with the collection of all the partitions of $[\![1, N]\!]$, whose number asymptotically behaves like $\mathcal{O}(Ne^N/\ln N)$, rendering the minimization of the criterion (4) computationally challenging. A way to bypass this issue for the change point only detection problem is via dynamic programming (Harchaoui & Cappé, 2007); this approach works in this simplified setup since there is a natural ordering for exploring the subproblems, which does not hold here. To overcome this, we will relax the criterion in such a way to create a subproblem ordering and thus derive a computationally feasible approximation. The proposed new method is outlined in Algorithm 1.

Let $\bar{y}_{[k]} := (\sum_{i\in[k]} Y_i)/|[k]|$, the average of the elements of Y in the $[k]$-th part. Notice that, given $\pi_m := \{[1]; [2]; \ldots; [d'_m - 1]\}$, we have

$$\mathbb{P}_{\mathcal{F}_m} Y = \sum_{k=1}^{d'_m-1} \frac{\langle Y, \mathbb{1}_{[k]}\rangle}{\|\mathbb{1}_{[k]}\|^2} \mathbb{1}_{[k]} = \sum_{k=1}^{d'_m-1} \bar{y}_{[k]} \mathbb{1}_{[k]}.$$

The minimization of criterion (4) can then be equivalently written as

$$\begin{aligned} &\min_{m\in\mathcal{M}} \mathrm{Crit}(m) \\ &= \min_{m\in\mathcal{M}} \{\|y - \mathbb{P}_{\mathcal{F}_m} Y\|_2^2 + \sigma^2 K \,\mathrm{pen}(d'_m, d''_m)\} \\ &= \min_{\substack{0\leqslant d' \\ \leqslant d'' \leqslant D}} \left\{ \min_{\substack{|m|=d' \\ |m|_0=d''}} \|y - \mathbb{P}_{\mathcal{F}_m} Y\|_2^2 + \sigma^2 K \,\mathrm{pen}(d', d'') \right\}, \end{aligned}$$

Algorithm 1 Two-Pass Dynamic Programming Algorithm

input data points $(y_i)_{i=1}^N$, maximum number of changes D and penalty strength K.

$$\bar{y}_{[k,l]} := \frac{\sum_{i=k}^{l} Y_i}{k-l+1}$$

$$R_{[k,l]} := \sum_{i=k}^{l} (y_i - \bar{y}_{[k,l]})^2, \quad 1 \leqslant k \leqslant l \leqslant N.$$

for $d = 1$ **to** D **do**

use the dynamic programming recurrence in (9) and a backtracking step to compute

$$C_d(N) := \min_{|\tilde{m}|=d} \|Y - \mathbb{P}_{\mathcal{F}_{\tilde{m}}} Y\|^2, \tag{5}$$

$$\tilde{m}_d \in \arg\min_{|\tilde{m}|=d} \|Y - \mathbb{P}_{\mathcal{F}_{\tilde{m}}} Y\|^2.$$

end for

for $d = 1$ **to** D **do**

$$\tilde{m}_d =: \{0 \leqslant i_1 < i_2 < \cdots < i_d < N\}$$

$$(\alpha_k)_{k=0}^d := (i_{k+1} - i_k)_0^d, \quad (i_0 = 0, i_{d+1} = N).$$

sort $(\bar{y}_{[1,i_1]}, \bar{y}_{[i_1+1,i_2]}, \ldots, \bar{y}_{[i_d+1,N]})$.

$\big(\bar{y}_{(k)}\big)_{k=0}^d$:= ordered sequence of $(\bar{y}_{[i_k+1,i_{k+1}]})_{k=0}^d$
$(\alpha_{(k)})_{k=0}^d$:= corresponding permuted $(\alpha_k)_{k=0}^d$ according to permutation ϕ_d.

$\bar{\bar{y}}_{(k,l)} := \frac{\sum_{i=k}^{l} \alpha_{(i)} \bar{y}_{(i)}}{\sum_{i=k}^{l-1} \alpha_{(i)}}$ and $\bar{R}_{[k,l]} := \sum_{i=k}^{l} \alpha_{(i)} (\bar{y}_{(i)} - \bar{\bar{y}}_{(k,l)})^2$, $1 \leqslant k \leqslant l \leqslant d$.

for $\delta = 1$ **to** d **do**

use the dynamic programming recurrence in (10) and a backtracking step to compute

$$G_{(d,\delta)} := \min_{m \in \mathcal{M}_{\bar{y}_{\tilde{m}},\delta}} \| \mathbb{P}_{\mathcal{F}_{\tilde{m}}} Y - \mathbb{P}_{\mathcal{F}_m} \mathbb{P}_{\mathcal{F}_{\tilde{m}}} Y\|^2, \tag{6}$$

$$\tilde{\tilde{m}}_{(d,\delta)} \in \arg\min_{m \in \mathcal{M}_{\bar{y}_{\tilde{m}}\delta}} \| \mathbb{P}_{\mathcal{F}_{\tilde{m}}} Y - \mathbb{P}_{\mathcal{F}_m} \mathbb{P}_{\mathcal{F}_{\tilde{m}}} Y\|^2.$$

end for

end for

$B_{(d,\delta)} := C_d + G_{(d,\delta)} + \sigma^2 K \operatorname{pen}((d,\delta))$, $1 \leqslant \delta \leqslant d \leqslant D$.

$(\hat{d}, \hat{\delta}) := \arg\min_{1 \leqslant \delta \leqslant d \leqslant D} B_{(d,\delta)}$.

reconstruct $m_{(\hat{d},\hat{\delta})}$ from $\tilde{m}_{\hat{d}}$ and $\tilde{\tilde{m}}_{(\hat{d},\hat{\delta})}$.

output value of criterion $\operatorname{Crit}(m_{(\hat{d},\hat{\delta})}) = B_{(\hat{d},\hat{\delta})}$ and selected model for change points and clusters $m_{(\hat{d},\hat{\delta})}$.

where D is a reasonable upper bound on the number of change points. As we shall see later, from a statistical point of view there is no need to explore all possible values of d' and d'', since the statistical guarantees only hold in a regime where $d' \leqslant d'' = o(N/\ln N)$.

We define $\pi_{\tilde{m}}$ to be the partition having as elements all the segments of π_m and instead of computing the minimum exactly we will take a greedy step by defining

$$\tilde{m} := \arg\min_{|\tilde{m}|=d''} \|Y - \mathbb{P}_{\mathcal{F}_{\tilde{m}}} Y\|^2$$

and defining $\mathcal{M}_{\tilde{m},d'} := \{m \in \mathcal{M} \colon m \subset \tilde{m}, |m| = d'\}$, which can be identified with the collection of all partitions of $[\![1, d'']\!]$ into d' sets. We restrict further this collection to partitions π satisfying what we call the **clustering property**, which states that if $\mathbb{I}_1$, $\mathbb{I}_2$ and $\mathbb{I}$ are segments in some (possibly different) parts of π, then

$$\begin{cases} \mathbb{I}_1, \mathbb{I}_2 \in [k] \\ \bar{y}_{\mathbb{I}_1} \leqslant \bar{y}_{\mathbb{I}} \leqslant \bar{y}_{\mathbb{I}_2} \end{cases} \quad \Rightarrow \quad \mathbb{I} \in [k]. \tag{7}$$

This sub-collection will be denoted as $\mathcal{M}_{\bar{y}_{\tilde{m}},d'}$. Simply put, this property says that the partitions considered are those that respect the ordering of $(\bar{y}_{[i_k+1,i_{k+1}]})_{k=0}^{d''}$, since if two segments $\mathbb{I}_1, \mathbb{I}_2$ belong to $[k]$, and the segment $\mathbb{I}$ satisfies $\bar{y}_{\mathbb{I}_1} \leqslant \bar{y}_{\mathbb{I}} \leqslant \bar{y}_{\mathbb{I}_2}$, then it should also be in cluster $[k]$.

This leads to the following upper bound, whose detailed derivation is given in appendix B:

$$\begin{aligned} \min_{m \in \mathcal{M}} \operatorname{Crit}(m) \leqslant \min_{0 \leqslant d'' \leqslant D} \Bigg\{ & \min_{|m|=d''} \|Y - \mathbb{P}_{\mathcal{F}_m} Y\|^2 \\ & + \min_{\substack{0 \leqslant d' \leqslant d'' \\ m \in \mathcal{M}_{\bar{y}_{\tilde{m}},d''}}} \| \mathbb{P}_{\mathcal{F}_{\tilde{m}}} Y - \mathbb{P}_{\mathcal{F}_m} \mathbb{P}_{\mathcal{F}_{\tilde{m}}} Y\|^2 \\ & + \sigma^2 K \operatorname{pen}(d', d'') \Bigg\}. \end{aligned}$$

Therefore, we can define the following relaxation for the minimization of the criterion in (4):

$$\begin{aligned} \operatorname{Crit}_r(d'') := & \min_{|m|=d''} \|Y - \mathbb{P}_{\mathcal{F}_m} Y\|^2 \\ & + \min_{\substack{0 \leqslant d' \leqslant d'' \\ m \in \mathcal{M}_{\bar{y}_{\tilde{m}},d''}}} \Bigg\{ \| \mathbb{P}_{\mathcal{F}_{\tilde{m}}} Y - \mathbb{P}_{\mathcal{F}_m} \mathbb{P}_{\mathcal{F}_{\tilde{m}}} Y\|^2 \\ & + \sigma^2 K \operatorname{pen}(d'_m, d''_m) \Bigg\}. \end{aligned} \tag{8}$$

and our algorithm computes $\min_{0 \leqslant d'' \leqslant D} \operatorname{Crit}_r(d'')$ and returns $m_{(\hat{d},\hat{\delta})}$. From this last definition we observe that

$$\min_{m \in \mathcal{M}} \operatorname{Crit}(m) \leqslant \operatorname{Crit}(m_{(\hat{d},\hat{\delta})}) = \min_{0 \leqslant d'' \leqslant D} \operatorname{Crit}_r(d'').$$

Thus, obtaining $m_{(\hat{d},\hat{\delta})}$ ensures making progress toward the minimization of $\mathrm{Crit}(m)$. The Two-Pass Dynamic Programming Algorithm 1 is aimed at doing this by computing the value of the minimum in (8) and returning a solution $\hat{m} = m_{(\hat{d},\hat{\delta})}$ in the following way:

Details of Main Steps in Algorithm 1

- **Step 3:** It computes $C_d(n)$ defined in (9) for all d and n to obtain $C_d(N)$ for all $d \in [\![1, N]\!]$. It does so by using a dynamic programming algorithm that computes recursively for all $2 \leqslant d \leqslant D$ and $d \leqslant n \leqslant N$ the following recurrence, similar to the one in Hawkins (1976):

$$\begin{aligned} C_1(n) &:= R_{[1,n]} \\ C_d(n) &:= \min_{i \in [\![d,n]\!]} \{C_{d-1}(i-1) + R_{[i,n]}\},\ d \geqslant 2. \end{aligned} \tag{9}$$

- **Step 7:** For all values of d, it sorts the obtained segments according to their levels to yield $\left(\bar{y}_{(k)}\right)_0^d$, and it keeps track of the segments' sizes as $(\alpha_k)_{k=0}^d = (i_{k+1} - i_k)_0^d$.

- **Step 11:** It runs a modified dynamic programming recurrence on $\left(\bar{y}_{(k)}\right)_0^d$ that uses weights according to the sizes $(\alpha_{(k)})_0^d$. It does so using the following recurrence for all $1 \leqslant \delta \leqslant t \leqslant d$:

$$\begin{aligned} G_{(t,1)} &:= \bar{R}_{[1,t]}, \\ G_{(t,\delta)} &:= \min_{i \in [\![\delta,t]\!]} \{G_{(i-1,\delta-1)} + \bar{R}_{[i,t]}\},\ \delta \geqslant 2. \end{aligned} \tag{10}$$

- **Step 15:** It computes the minimum in (8) and finds for which model it is attained by solving the minimization problem:

$$(\hat{d}, \hat{\delta}) := \arg \min_{1 \leqslant \delta \leqslant d \leqslant D} B_{(d,\delta)}.$$

- **Step 16:** It finally reconstructs $m_{(\hat{d},\hat{\delta})}$ from $\tilde{m}_{\hat{d}}$ and $\tilde{\tilde{m}}_{(\hat{d},\hat{\delta})}$ using the permutation $\phi(\hat{d})$.

This algorithm can be thought of as an efficient way to compute the relaxation in (8), based on solving the change point detection problem in (5) using the dynamic programming recurrence of (9), followed by a solving a clustering problem in (6) using the dynamic programming recurrence of (10).

The next theorem shows that Algorithm 1 correctly solves the minimization problem in (8) and explicits its time and space complexity.

Theorem 4.1. *Let $(y_i)_{i=1}^N \subset \mathbb{R}$, $D \in \mathbb{N}$ and $K > 0$. Then,*

- *for all $1 \leqslant d \leqslant D$,*

$$\tilde{m}_d \in \arg \min_{|\bar{m}|=d} \|Y - \mathbb{P}_{\mathcal{F}_{\bar{m}}} Y\|^2,$$

- *for all $1 \leqslant \delta \leqslant d \leqslant D$,*

$$\tilde{\tilde{m}}_{(d,\delta)} \in \arg \min_{m \in \mathcal{M}_{\bar{y}_{\tilde{m}}\,\delta}} \| \mathbb{P}_{\mathcal{F}_{\tilde{m}}} Y - \mathbb{P}_{\mathcal{F}_m} \mathbb{P}_{\mathcal{F}_{\tilde{m}}} Y\|^2.$$

Furthermore, Algorithm 1 correctly solves the minimization problem in (8), with time and space complexity $\mathcal{O}(N^3 + D^4)$ and $\mathcal{O}(N^2 + D^3)$, respectively.

Proof. See Appendix B. □

The time and space complexity can be improved to $\mathcal{O}(N^2 D + D^4)$ and $\mathcal{O}(DN + D^3)$, respectively. We refer the reader to the discussion after the proof in Appendix B for the derivation of this result. In this way we obtain a computationally feasible algorithm that finds the minimum in (8) and returns an approximation to the criterion in (4). In the next section, we will motivate the use of Algorithm 1 from a statistical point of view by showing that the minimization of criterion (4) can be viewed as an approximate maximum a-posteriori estimator.

5. Model selection criterion for change point detection and clustering

In this part, we provide a derivation of the optimization criterion in (4). We start by proposing a Bayesian model selection scheme, which is later inverted to arrive at an integral form of the maximum a-posteriori probability (MAP) estimator. Then we use a Laplace approximation to derive turn the MAP into an optimization problem of the desired form.

Here we show that the proposed selection criterion in (4) follows naturally from a Bayesian reasoning. For this, we model the data as being the outcome of the following sampling model. The observation Y is generated from a multivariate Gaussian of mean F and variance $\sigma^2 I_N$ as described by (1). For the random variable F, given that it belongs to a subspace $\mathcal{F}_m$, we choose an absolutely continuous measure $\mathcal{L}^{d'_m}$ with respect to $\lambda^{d'_m}$, the Lebesgue measure on $\mathbb{R}^{d'_m+1}$, such that $d\mathcal{L}^{d'_m} = l_{f/m} d\lambda^{d'_m+1} = \prod_{k=1}^{d'_m+1} (l_{f_k/m} d\lambda)$ with $l_{f_1/m} = \cdots = l_{f_{d'_m+1}/m}$. Later we will see that the choice $l_{f/m}$ will not matter in comparison to the order of approximation, nevertheless we would like it to be a bounded continuous prior satisfying some additional conditions given in Lemma 2, even though we might be chosen as an improper prior. On the family of models $\mathcal{M}$ we impose a categorical distribution measure $\mathbb{P}_{\mathcal{M}}$ as prior, with a weight p_m for model m. Thus, we obtain the following sampling model for the data[1]:

$$\begin{aligned} Y/F &\sim \mathcal{N}(F, \sigma^2 I_N) \\ F/m &\sim \mathcal{L}^{d'_m} \\ m &\sim \mathbb{P}_{\mathcal{M}} = \mathrm{Categorical}((p_m)_{m \in \mathcal{M}}). \end{aligned} \tag{11}$$

Since Y, F and m are now random variables, it makes sense to compute $\mu_{m/Y}$, the posterior distribution of m

[1] Here and in the sequel, the dependence of p_m and $\mathbb{P}_{\mathcal{M}}$ on the number of samples N is omitted, for simplicity of notation.

given Y, and maximize it, to arrive at a MAP estimate of m given bellow.

$$p_{m/Y} = \frac{p_m \int_{f\in\mathcal{F}_m} \phi_N\left(\frac{Y-f}{\sigma}\right) l_{f/m}(f)df}{\sum_{m'\in\mathcal{M}} p_{m'} \int_{f'\in\mathcal{F}_m} \phi_N\left(\frac{Y-f'}{\sigma}\right) l_{f/m'}(f')df'}. \tag{12}$$

For the complete derivation of the formula in 12 we refer you to appendix B.

Starting from the a-posteriori distribution (12) we can derive an approximation for the MAP as follows:

$$\begin{aligned} p_{m/Y} &\propto p_m \int_{f\in\mathcal{F}_m} \phi_N\left(\frac{Y-f}{\sigma}\right) l_{f/m}(f)df \\ &= p_m \prod_{k=1}^{d'_m+1} \frac{1}{(2\pi\sigma^2)^{\frac{|[k]|}{2}}} \\ &\quad \cdot \int_{\mathbb{R}} \exp\left(-\frac{\|y_{[k]} - f_k \mathbb{1}_{[k]}\|_2^2}{2\sigma^2}\right) l_{f_k/m}(f_k)df_k. \end{aligned} \tag{13}$$

In the last step of (13) we define $y_{[k]}$ as the vector obtained from the entries of y corresponding to cluster $[k]$. To obtain an approximation of the MAP estimate as a solution of a criterion of the form (4) we need the result of lemma 2 stated and proved in Appendix C using a Laplace approximation type of argument. We then obtain the following upper bound for the MAP for all $K \geqslant 1$:

$$\begin{aligned} \mathrm{Crit}_{\mathrm{MAP}}(m) &\leqslant \frac{\|y - \mathbb{P}_{\mathcal{F}_m} y\|_2^2}{2\sigma^2} \\ &+ K\left(\ln\frac{1}{p_m} + \frac{1}{2}(d'_m+1)\ln\frac{N}{d'_m}\right) + \mathcal{O}(d'_m). \end{aligned} \tag{14}$$

The complete derivation of (14) can be found in Appendix C. Now we define our approximate MAP criterion as:

$$\begin{aligned} \mathrm{Crit}(m) &= \|y - \mathbb{P}_{\mathcal{F}_m} y\|_2^2 + \sigma^2 K\,\mathrm{pen}(m), \\ \mathrm{pen}(m) &= \left(2\ln\frac{1}{p_m} + (d'_m+1)\ln\frac{N}{d'_m}\right). \end{aligned} \tag{15}$$

In the next section, we finish the specification of the penalty term by providing the probabilities p_m over the space of models. To do so we will exhibit an oracle inequality satisfied by the estimator that minimizes (4), and choose a probability mass function (p_m) that gives a reasonable upper bound on the expected quadratic risk defined below.

6. Oracle inequality and upper bound for the risk

The standard way of assessing the performance of a statistical algorithm is by comparing its performance to a reasonable oracle. For this we use as a measure of performance of an estimator $\hat{f}$ the expected quadratic risk:

$$\mathcal{R}_n(\hat{f}) = \mathbb{E}[\|\hat{f} - f^*\|_2^2].$$

In the case of the change point detection and clustering problem, the comparison should be non-asymptotic, reflecting our lack of knowledge about both the clustering dimension and the change point dimension. For this we state below a non-asymptotic oracle inequality for $\mathrm{Crit}(m)$ using an oracle with remainder of the form:

$$\inf_{m\in\mathcal{M}}\{\mathcal{R}_n(\mathbb{P}_{\mathcal{F}_m} y) + o_m(1)\}.$$

This type of oracle has access to f^* and chooses the m that minimizes the risk criterion up to a remainder term.

To derive this we finish the specification of $\mathrm{Crit}(m)$ by providing an appropriate prior p_m. The intuition behind our choice is the following. Defining $\hat{r}_m = \|y - \hat{f}_m\|_2^2$ and $\mathrm{pen}(m) = 2\sigma^2\ln\frac{1}{p_m} + \sigma^2(d'_m+1)\ln\frac{N}{d'_m}$ we see that the criterion (15) is of the form:

$$\mathrm{Crit}(m) = \hat{r}_m + \mathrm{pen}(m).$$

The number of models in the family $\mathcal{M}$ having the same values of d'_m and d''_m grows exponentially with those dimensions. Thus for fix d'_m and d''_m we might find a model with low $\hat{r}_m$ just because of randomness since some of them will deviate largely from their means, which would correspond to an over-fitting case, this was the problem of case with the traditional AIC type of estimators. Therefore, we need to penalize models of high dimensions more by taking into account the number of models with same dimensions. On the other hand we want this penalty to be as small as possible this way we give more importance to the fitting term $\hat{r}_m$. In particular we would prefer the term $2\sigma^2\ln\frac{1}{p_m}$ to stay close to $\sigma^2(d'+1)\ln\frac{N}{d'}$ at least for values of d'_m close to d''_m. Our choice for p_m, useful inequalities and a complete discussion of the role of p_m as a prior and tuning parameter for the risk can be found in Appendix D. From Lemmas 3 and 4, the following oracle inequality can be derived for $\hat{f}_{\hat{m}}$:

Theorem 6.1 (Oracle inequality for $\hat{f}_{\hat{m}}$)**.** *With $\mathcal{M}$ restricted to models such that $ed'_m \leqslant N$ and for the choice of $K = 3a$, p_m as in 3, $\mathrm{pen}(m)$ as in 15 and $\hat{m} \in \mathcal{M}$ corresponding to*

$$\hat{m} \in \arg\min_{m\in\mathcal{M}} \|y - \hat{f}_m\|_2^2 + \sigma^2 K\,\mathrm{pen}(m), \tag{16}$$

We obtain for all $a > 1$,

$$\begin{aligned} E_{f^*}[\|\mathbb{P}_{\mathcal{F}_{\hat{m}}} Y - f^*\|^2] \leqslant \\ \arg\min_{m\in\mathcal{M}} \left\{ \frac{a}{a-1}\mathbb{E}_{f^*}[\|\mathbb{P}_{\mathcal{F}_m} Y - f^*\|^2] \right. \\ \left. + \frac{a^2\sigma^2}{a-1}\left(7 + 3(d'_m+1)\ln\frac{N}{d'_m} + 6\ln\frac{1}{p_m}\right)\right\}. \end{aligned} \tag{17}$$

Proof. See Appendix D. □

By investigating the oracle inequality, one notices that for an optimal choice of a one has to make a trade-off between the performance of the oracle part and the bias part of the inequality. In general this trade-off is not possible to optimize since the value of the oracle part is not available to us and depends on the variance of the noise. In practice, one can use the SLOPE heuristic introduced in Lebarbier (2002) and described in Baudry et al. (2012) and in (Arlot & Massart, 2009). In our case, the value of the tuning parameter can be chosen independently of the variance of the noise and we can use the value of a for which we know that our estimator $\hat{f}_{\hat{m}}$ will perform well.

Corollary 6.1. *For the set of models described in 6.1 with $f^* \in \mathcal{F}_{m^*}$ the following properties hold:*

- *Adaptation and Risk Upper bound: The following adaptive upper bound in terms of d'_{m^*} and d''_{m^*} holds for $a = 2$:*

$$E_{f^*}[\| \mathbb{P}_{\mathcal{F}_{\hat{m}}} Y - f^* \|^2] \leqslant 4\sigma^2 \Bigg(7 + 3(d'_{m^*} + 1) \ln \frac{N}{d'_{m^*}} + 6 \Bigg(d'_{m^*} \ln[d''_{m^*} e^{\frac{13}{6}}] + d''_{m^*} \ln[d'_{m^*} e^2] + d''_{m^*} \ln \frac{N}{d''_{m^*}} \Bigg) \Bigg).$$

- *Consistency: If $d''_{m^*} = o(N/\ln N)$, then $\lim_{N\to\infty} N^{-1} \mathbb{E}_{f^*}[\| \hat{f}_{\hat{m}} - f^* \|^2] = 0$.*

Proof. See Appendix D. □

We notice that the consistency condition $d''_{m^*} = o(N/\ln N)$ is within the restriction on the models in theorem 6.1, hence there is no loss of generality of having only models with $e d'_m \leqslant N$ in $\mathcal{M}$ since for other models we cannot guarantee convergent mean square risk anyway. In the special case $d'_{m^*} = d''_{m^*}$, i.e when the change point and clustering problem reduces to a change point only problem, Kernel methods have comparable accuracy (Celisse et al., 2017). The interesting case is when the numbers are different, we gain a logarithmic factor in accuracy with almost the same computational cost. In the next section, we validate these theoretical guarantees by a series of tests on simulated data to get a sense of how tight the oracle inequality is, which signals are difficult to estimate and how the algorithm behaves in practice.

7. Experimental results

Consider first an experiment based data generated randomly according to the setup of (1) with the same change points of Example 1. This is considered to be an easy case since $d'_{m^*} = 4 < d''_{m^*} = 12 \ll N = 2000$, which is within the range of signals for which the consistency result of Corollary 6.1 holds.

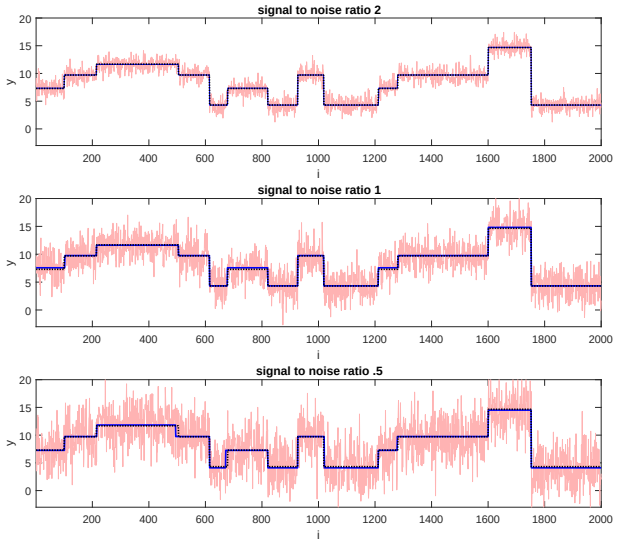

Figure 2. Estimates $\hat{f}$ (blue line) of f^* (dotted black line) obtained by Algorithm 1 using the observed signal Y (pink line), with 3 different levels of signal-to-noise ratio.

The experiments in Figure 2 show that the algorithm is quite robust to the level of noise as measured by the signal-to-noise ratio $S/N = \frac{\text{magnitude of smallest jump in } f^*}{\sigma^2}$. We observe that the difference between the ground truth f^* and $\hat{f}_{\hat{m}}$ is quite small even for small S/N levels such as $S/N = 0.5$ and the change point locations do not vary appreciably; in fact, for this experiment, $S/N = 0.3$ seems to be the limiting case for which the algorithm performs well, and for lower values the risk upper-bound in Corollary 6.1 becomes loose when σ increases. Also, we note that an S/N of 0.5 is quite low for this kind of problems. In particular, algorithms relying on the L_1-penalty such as Fussed LASSO do not achieve this kind of performance on the simpler task of change point only detection, while on the other hand, they are more computational efficient (Xin et al., 2014).

Figure 3 illustrates a difficult case, where we reduced the number of observation by segment by scaling down the signal f^* to a support of size $N = 500$. Now we are outside of the useful regime of Corollary 6.1 and we notice that the second segment $[\![15, 53]\!]$ is wider than what it should since the first change point at 25 was detected at 14; also the segment $[\![206, 237]\!]$ belongs to cluster $[4]$ while it is actually in cluster $[3]$ in the original signal f^*. Nevertheless we can observe an interesting property for segment $[\![324, 346]\!]$, namely, that the end point 346 does not correspond to any real change point, yet this segment belongs to the optimal solution of the 1^{st} dynamic programming pass. On the other hand the 2^{nd} dynamic programming pass puts it in the same cluster $[3]$ as $[\![347, 399]\!]$, turning them into one single segment of cluster $[3]$. This behavior actually is the norm for the algorithm, where false changes are often detected in difficult signals in the 1^{st}

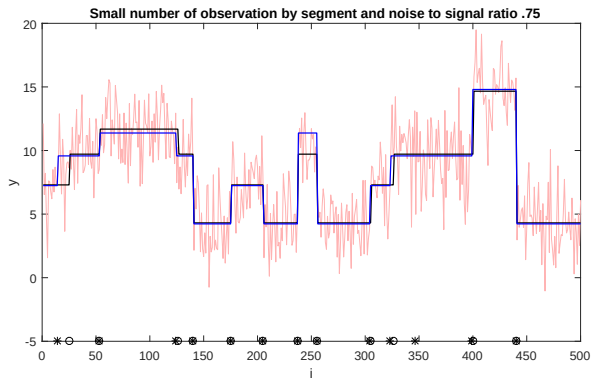

Figure 3. Estimate $\hat{f}$ (blue line) of f^* (black line) obtained by Algorithm 1 from a difficult observation sample Y (pink line) with high signal-to-noise ratio (1.5) and few observations per segment ($N = 500$ and $d''_{m^*} = 13$).

dynamic programming pass but are removed after the 2^{nd} pass. These kinds of false discoveries are actually one of the weaknesses of many change point only detection algorithms like Fussed LASSO, and they have been studied in (Levy-leduc & Harchaoui, 2008), (Rinaldo, 2009) and (Rojas & Wahlberg, 2014). In the last experiment,

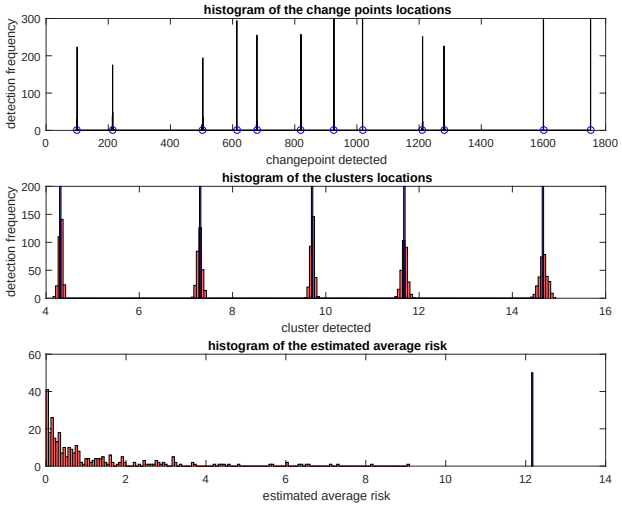

Figure 4. Top histogram: location of estimated (black) and true (black) change points in 300 simulations. Middle histogram: location of estimated (red) and true (blue) clusters in 300 simulations. Bottom histogram: 300 estimates of the average mean square error (red) and its theoretical upper bound (blue).

we run Algorithm 1 300 times with the parameter values $d'_{m^*} = 4 < d''_{m^*} = 12 \ll N = 2000$ and signal-to-noise ratio $S/N = 1$; Figure 4 summarizes the results. In the top histogram we notice that the algorithm successfully detects the change points most of the time; in fact, the achieved accuracy was $\frac{\text{number of change points correctly detected}}{\text{number of change points detected}} \approx 0.8528$. The middle histogram shows the placement of estimated clusters and the true values of the clusters; we observe that the true values lie in a small neighborhood of the estimated values for every cluster. In the bottom histogram we observe that the theoretical upper bound on the average mean square error –in this case 12.1575– found in Corollary 6.1 is very conservative and most of the 300 estimates –given by $\frac{\|\hat{f}_{\hat{m}} - f^*\|^2}{N}$– are significantly smaller.

8. Conclusions

In this work, we considered a novel problem related to change point detection where we have to address the simultaneous task of segmenting and clustering the observed signal. Our approach has been to view this problem as a non-parametric model selection problem on the set of all possible partitions. We derived for this the computationally tractable Algorithm 1, that computes a relaxation of the penalized minimization of criterion (4), and we justified it from a statistical standpoint by showing that this minimization can be viewed as an approximate MAP. This approximate MAP estimate enjoys the properties of being adaptive and consistent in the sense of Corollary 6.1. We finally justified the use of Algorithm 1 by simulation data that shows some useful properties of the resulting estimate and validates the theoretical guarantees.

One extension of this work concerns developing a more complete analysis of Algorithm 1, to obtain consistency results on the number and locations of the change points and clusters. Another possible extension relates to the use of Algorithm 1 in the non-scalar case; this was already explored for change point only detection in (Arlot et al., 2016) through the use of characteristic kernels (Sriperumbudur et al., 2011). We believe that the same approach can be adopted here except that we cannot perform the sorting step; this can be overcome using a Kernel clustering algorithm (Filipponea et al., 2008) or a spectral version of it (Schölkopf et al., 1998) for the second stage. Finally, the remark after Figure 3 hints to the possibility of using a combined algorithm starting with the sparse solution of Fussed LASSO and running the 2^{nd} dynamic programming pass of our algorithm as a way to boost the performance of Fussed LASSO to get rid of false discoveries. This would be still computationally attractive according to the comment after Theorem 4.1, since the solution of Fussed LASSO has a small number of changes.

References

Abou-Elailah, A., Gouet-Brunet, V., and Bloch, I. Detection of abrupt changes in spatial relationships in video sequences. In *International Conference on Pattern Recognition Applications and Methods*, pp. 89–106, 2016.

Arlot, S. and Massart, P. Data-driven calibration of penalties for least-squares regression. *Journal of Machine Learning Research*, 10:245–279, 2009.

Arlot, S., Celisse, A., and Harchaoui, Z. Kernel change-point detection. *ArXiv:1202.3878v1*, 2016.

Arnold, T. B. and Tibshirani, R. J. Efficient implementations of the generalized lasso dual path algorithm. *Journal of Computational and Graphical Statistics*, 25(1): 1âĂŞ–27, 2016.

Baudry, J. P., Maugis, C., and Michel, B. Slope heuristics: overview and implementation. *Statistics and Computing*, 22:455âĂŞ–470, 2012.

Boucheron, S., Lugosi, G., and Massart, P. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 2013.

Celisse, A., Marot, G., Pierre-Jean, M., and Rigaill, G. New efficient algorithms for multiple change-point detection with kernels. *arXiv:1710.04556*, 2017.

Cirel'son, B. S., Ibragimov, I. A., and Sudakov, V. N. Norms of Gaussian sample functions. In *Proceedings of the Third Japan-USSR Symposium on Probability Theory*, pp. 20–41, 1976.

Cleynen, A. and Lebarbier, E. Segmentation of the Poisson and negative binomial rate models: a penalized estimator. *Probability and Statistics*, 18(2):750âĂŞ–769, 2014.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (eds.). *Introduction to Algorithms, Third Edition*. MIT Press, Cambridge, MA, USA, 2009.

Dalalyan, A. S., Hebiri, M., and Lederer, J. On the prediction performance of the Lasso. *Bernoulli*, 23(1):552–581, 2017.

Filipponea, M., Camastrab, F., Masullia, F., and Rovetta, S. A survey of kernel and spectral methods for clustering. *Journal of Machine Learning Research*, 41(1):176–190, 2008.

Garreau, D. and Arlot, S. Consistent change-point detection with kernels. *arXiv:1612.04740*, 2017.

Goodrich, M. T. and Tamassia, R. (eds.). *Algorithm Design: Foundations, Analysis, and Internet Examples*. John Wiley and Sons, Hoboken, NJ, USA, 2001.

Graham, R. L., Knuth, D. E., and Patashnik, O. *Concrete Mathematics*. AddisonâĂŞWesley, 1988.

Harchaoui, Z. and Cappé, O. Retrospective multiple change-point estimation with kernels. In *14th IEEE/SP Workshop on Statistical Signal Processing (SSP '07)*, Madison, WI, USA, 2007.

Hawkins, Douglas M. Point estimation of the parameters of piecewise regression models. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 25(1): 51–57, 1976. ISSN 00359254, 14679876. URL http://www.jstor.org/stable/2346519.

Hütter, J. and Rigollet, P. Detection of abrupt changes in spatial relationships in video sequences. In *29th Annual Conference on Learning Theory, PMLR 49*, pp. 1115–1146, 2016.

Kim, A. Y., Marzban, C., Percival, D. B., and Stuetzle, W. Using labeled data to evaluate change detectors in a multivariate streaming environment. *Signal Processing*, 89 (12):2529âĂŞ–2536, 2009.

Lai, W. R., Johnson, M. D., Kucherlapati, R., and Park, P. J. Comparative analysis of algorithms for identifying amplifications and deletions in array CGH data. *Bioinformatics*, 21:3763–âĂŞ3770, 2005.

Lavielle, M. and TeyssiÃĺre, G. Detection of multiple change-points in multivariate time series. *Lithuanian Mathematical Journal*, 46:287–âĂŞ306, 2006.

Lebarbier, E. *Quelques approches pour la d'etection de ruptures a horizon fini*. PhD thesis, FacultÃl' des Sciences d'Orsay (Essonne), Universite Paris-Sud, 2002.

Levy-leduc, C. and Harchaoui, Z. Catching change-points with lasso. In *Advances in Neural Information Processing Systems 20*, pp. 617–624, 2008.

Massart, P. *Concentration Inequalities and Model Selection*. Springer-Verlag Berlin Heidelberg, 1st edition, 2003.

Mohr, A. and Porter, T. D. Applications of chromatic polynomials involving Stirling numbers. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 70:57–âĂŞ64, 2009.

O., Wittich, A., Kempe, G., Winkler, and V., Liebscher. Complexity penalized least squares estimators: Analytical results. *Mathematische Nachrichten*, 281 (4):582–595. doi: 10.1002/mana.200510627. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/mana.200510627.

Rennie, B. C. and Dobson, A. J. On Stirling numbers of the second kind. *Journal of Combinatorial Theory*, 7(2): 116–121, 1969.

Rigaill, G., Lebarbier, E., and Robin, S. Exact posterior distributions and model selection criteria for multiple change-point detection problems. *Statistics and Computing*, 22(4):917–âĂŞ929, 2012.

Rinaldo, A. Properties and refinements of the fused lasso. *The Annals of Statistics*, 37(5b):2922–âĂŞ2952, 2009.

Rojas, C. R. and Wahlberg, B. On change point detection using the fused lasso method. *arXiv:1401.5408*, 2014.

Rudin, L. I., Osher, S., and Fatemi, E. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–âĂŞ268, 1992.

Schölkopf, B., Smola, A., and Müller, K. R. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.

Spokoiny, V. Multiscale local change point detection with applications to value-at-risk. *The Annals of Statistics*, 37:1405–1436, 2009.

Sriperumbudur, B. K., Fukumizu, K., and Lanckriet, G. R. G. Universality, characteristic kernels and rkhs embedding of measures. *Journal of Machine Learning Research*, 12:2389–2410, 2011.

Tartakovsky, A., Nikiforov, I. V., and Basseville, M. *Hypothesis Testing and Changepoint Detection*. Chapman and Hall, Monographs on Statistics and Applied Probability, 2014.

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., and Knight, K. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91âĂŞ–108, 2005.

Xin, B., KawaharaâĂă, Y., Wang, Y., and Gao, W. Efficient generalized fused lasso and its application to the diagnosis of alzheimerâĂŹs disease. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.