Bioinformatics, 39(4), 2023, btad170
https://doi.org/10.1093/bioinformatics/btad170
Advance Access Publication Date: 11 April 2023
Original Paper

Genetics and population analysis

Population-based change-point detection for the identification of homozygosity islands

Lucas Prates[1], **Renan B. Lemes**[2], **Tábita Hünemeier** [2,3], **Florencia Leonardi** [1,*]

[1]Institute of Mathematics and Statistics, University of São Paulo, São Paulo, Brazil
[2]Institute of Biological Sciences, University of São Paulo, São Paulo, Brazil
[3]Institut de Biologia Evolutiva, Universitat Pompeu Fabra, Barcelona, Spain

*Corresponding author. Institute of Mathematics and Statistics, University of São Paulo, São Paulo, Brazil. E-mail: florencia@usp.br

Associate Editor: Russell Schwartz

Received 28 October 2022; revised 7 February 2023; accepted 14 March 2023

Abstract

Motivation: This work is motivated by the problem of identifying homozygosity islands on the genome of individuals in a population. Our method directly tackles the issue of identification of the homozygosity islands at the population level, without the need of analysing single individuals and then combine the results, as is made nowadays in *state-of-the-art* approaches.

Results: We propose regularized offline change-point methods to detect changes in the parameters of a multidimensional distribution when we have several aligned, independent samples of fixed resolution. We present a penalized maximum likelihood approach that can be efficiently computed by a dynamic programming algorithm or approximated by a fast binary segmentation algorithm. Both estimators are shown to converge almost surely to the set of change-points without the need of specifying *a priori* the number of change-points. In simulation, we observed similar performances from the exact and greedy estimators. Moreover, we provide a new methodology for the selection of the regularization constant which has the advantage of being automatic, consistent, and less prone to subjective analysis.

Availability and implementation: The data used in the application are from the Human Genome Diversity Project (HGDP) and is publicly available. Algorithms were implemented using the R software R Core Team (*R: A Language and Environment for Statistical Computing*. Vienna (Austria): R Foundation for Statistical Computing, 2020.) in the R package `blockcpd`, found at https://github.com/Lucas-Prates/blockcpd.

1 Introduction

In diploid organisms, such as humans, each individual's genome is organized into pairs of chromosomes, each half inherited from each parent. When an individual is an offspring of biologically related parents, both chromosomes of the same pair can share identical segments, creating long stretches of consecutive homozygosity, known as runs of homozygosity (ROH).

In the last decades, studies on the identification of ROH carried out in human populations have revealed the presence of ROH even in cosmopolitan non-inbred populations, disclosing an increment of inbreeding levels and the consequent reduction of genetic diversity of populations, which is proportional to the walking distance from Africa, as expected by the out-of-Africa model of human colonization (Kirin et al. 2010, Leutenegger et al. 2011, Pemberton et al. 2012, Ceballos et al. 2018, Lemes et al. 2018).

The distribution of ROH along the chromosomes is very uneven, resulting in some genomic regions having significant absence (coldspots) or excess of ROH (ROH islands) (Ceballos et al. 2018). The mechanisms for the emergence of these regions are still under discussion. For example, there is evidence that ROH islands could represent regions that harbor genes target of positive selection since low-recombination regions commonly are locations of selective sweeps, in which a new beneficial mutation increases in frequency and becomes fixed, causing the overall reduction in genetic diversity of the region (Pemberton et al. 2012, Ceballos et al. 2018).

To detect ROH and ROH islands, the genetic material of individuals from a given population is genotyped, and a set of single nucleotide polymorphisms (SNPs) is obtained. Each SNP entry is codified to 1 if that SNP belongs to an ROH for that individual and to 0 otherwise, where a marker is defined as belonging to an ROH for an individual if it is surrounded by a region with high frequency of homozygous SNPs. Finally, ROH islands are regions in which ROH are most frequent in that population. That is, the positions in the array with high frequency of individual ROH passing through them. Therefore, we can think on the problem of ROH island detection in a population as the identification of regions with high frequencies of 1's in the codified SNPs of individuals of that population. That is, this problem can be regarded as a change-point problem for the parameters of a multidimensional random vector with Bernoulli marginal distributions.

1.1 Change-point detection

Classically, change-point detection refers to the problem of determining the times at which sequential observed data undergoes an abrupt change. In that type of setting, a change-point may refer to changes in mean (Page 1954, Tsay 1988, Keshavarz et al. 2018), variance (Chen and Gupta 1997, Hawkins and Zamba 2005), regression slope (Chow 1960, Qu and Perron 2007), general distributions forms (Matteson and James 2014), or other types of change (Castro et al. 2018, Leonardi et al. 2021). Many of these methods have been applied to a wide range of problems such as stream anomaly detection in industry (Li et al. 2018), monitoring of sleep stages using EEG/EMG (Agudelo-España et al. 2020), identification of cyberattacks on networks (Tartakovsky et al. 2006), between many other interesting applications. For more references about change-point methods for time series and other applications, we refer the reader to the book Chen and Gupta (2012) or the reviews Truong et al. (2020) and Lee (2010).

As Niu et al. (2016) points out, from a methodological point of view, there is a variety of ways to formulate the change-point detection problem: online versus offline, single change-point versus multiple change-points, parametric versus nonparametric, Bayesian versus non-Bayesian, and many other approaches when we dive into specific estimation procedures. But in most of the classical formulations for change-point detection, the estimation problem is considered under the hypothesis that the number of observations along the dimension of interest grows. Early results of Hinkley (1970) show that, even for more straightforward problems, these methods are usually consistent for the change-points fractions, but not the change-points properly. For offline issues, a possible interpretation is that we have a fixed number of series (usually one) over a unit interval. What grows with the number of observations, in this case, is the resolution over this interval.

As technology advances and other types of data arise from different application areas, new types of change-point detection problems arise as well. As we will see, ROH islands detection for a population is an example of this. For this problem, the dimension of interest is discrete and finite, and what grows is the number of independent observations of the random vector. Moreover, since the indices correspond to specific positions such as biological markers, it is essential to recover the exact location of the change-points. Therefore, it is more appropriate to have theoretical consistency to the true change-points as the number of independent observations of the joint multidimensional distribution grows.

1.2 Our contribution and related works

This work considers several aligned, independent samples of a multidimensional distribution, and we assume the distribution has a block structure with different parameters on each block. We focus on detecting such changes in the parameters, that is, the boundaries of the blocks when the number of observations grows, without assuming the number of blocks is known *a priori*. We aim to jointly estimate the number and the position of the change-points in the sample.

We propose a penalized maximum likelihood approach to detect the set of change-points in the distribution. General distribution-free conditions for the consistency of the exact estimator using dynamic programming and the greedy efficient estimator using binary segmentation are presented. Indeed, proving the consistency of the binary segmentation algorithm is a complex problem in change-point theory, and we prove its consistency using techniques different from previous works by Rice and Zhang (2022), Fryzlewicz (2014) and Venkatraman (1992). To show that our framework is reasonable, we verify the conditions leading to the consistency of the estimators for standard distributions such as categorical random variables and Gaussian random variables. Still, other families of distributions can satisfy these conditions, widening the possibility of applying our method to many different applied problems.

Another interesting contribution to the theory is the introduction of the First Repeated Value (FRV) methodology to select the penalizing constant. The selection of the penalizing constant for this type of estimator is a complex problem in practice and can drastically impact its performance. Albeit reference methodologies as provided by Lavielle (2005) and Haynes et al. (2017) exist, they have drawbacks such as relying on visual inspection. Our methodology is fully automatic and still preserves the consistency of the algorithms.

We discuss the identification of the ROH islands in the African and European populations based on the analysis of SNP data from the Human Genome Diversity Project (HGDP) (Li et al. 2008). We compare the ROH islands detected by our method with those detected by the procedure proposed in McQuillan et al. (2008); Kirin et al. (2010), using the PLINK software (Purcell et al. 2007), a well-established command-line software designed to solve many medical and population genetics problems.

2 System and methods

Let m be a positive integer and consider a random vector $\mathbf{X} = (X_1, \ldots, X_m)$ taking values in A^m, where A is any state space. Let Θ be a parametric space and consider a family $\mathcal{F} = \{f_\theta | \theta \in \Theta\}$ of probability densities or probability mass functions over A^ℓ, for each $\ell \in \mathbb{N}$, indexed by $\theta \in \Theta$. We assume each variable X_i, $i = 1, \ldots, m$ is distributed according to some $f_\theta \in \mathcal{F}$. Given two integers r and s, with $r \leq s$, we use the notation $r : s$ for the set $\{r, r+1 \ldots, s\}$. Denote by $\mathcal{C}$ the class of all ordered sets $\mathrm{C} = \{c_0, c_1, \ldots, c_k\} \subseteq 0 : m$ such that $c_0 = 0$ and $c_k = m$. We say that $\mathrm{C} \in \mathcal{C}$ is a change-point set for the random vector $\mathbf{X}$ if the variables in the subvector $(X_{c_{j-1}+1}, \ldots, X_{c_j})$ are identically distributed with distribution $f_{\theta_j} \in \mathcal{F}$ and $\theta_j \neq \theta_{j+1}$ for all $j = 1, \ldots, k-1$. Observe that the change-point vector is unique, and from now on it will be denoted by C^*. Given any set $\mathrm{C} \in \mathcal{C}$, we denote by k_C the number of positive elements in C; that is, if $\mathrm{C} = \{c_0, c_1, \ldots, c_k\}$ then $k_\mathrm{C} = k$.

Assuming the blocks in the random vector $\mathbf{X}$ are independent, we can write the probability of observing $\mathbf{x} = (x_1, \ldots, x_m) \in A^m$ in the model with parameters (C, θ), $\theta = \{\theta_j \in \Theta : j \in 1 : k_\mathrm{C}\}$ by:

$$\mathbb{P}_{(\mathrm{C},\theta)}(\mathbf{x}) = \prod_{j=1}^{k_\mathrm{C}} f_{\theta_j}(x_{(c_{j-1}+1):c_j}), \tag{1}$$

where $f_{\theta_j}(x_{(c_{j-1}+1):c_j})$ represents the distribution of a random vector over $A^{c_j - c_{j-1}}$ with parameter θ_j. The independence assumption over the different blocks is not a necessary condition for the method but the generalization to a nonindependent setting is out of the scope of this work.

Consider a sample of n i.i.d. random vectors $\mathbf{x}^n = \{\mathbf{x}^{(i)}\}_{i=1}^n$ distributed as $\mathbf{X}$, with change-point set $\mathrm{C}^* = (c_0^*, \ldots, c_{k^*}^*)$ and parameters $\theta^* = (\theta_1^*, \ldots, \theta_{k^*}^*)$. Our main goal is to estimate the change-point set C^* and the parameters θ^*.

For any integer interval $I \subset 1 : m$ assume, we can compute the maximum likelihood estimator based on the subsample $\{x_j^{(i)}\}_{i \in 1:n, j \in I}$. Write the maximum likelihood function for the set C of candidate change-points as:

$$L(\mathrm{C}; \mathbf{x}^n) = \prod_{i=1}^n \mathbb{P}_{(\mathrm{C},\hat{\theta})}(\mathbf{x}^{(i)}) = \prod_{i=1}^n \prod_{j=1}^{k_\mathrm{C}} f_{\hat{\theta}_j}\left(x_{(c_{j-1}+1):c_j}^{(i)}\right) \tag{2}$$

where $\hat{\theta}_j$ denotes the maximum likelihood estimator computed on the sample $\{x_c^{(i)}\}_{i \in 1:n, c \in I_j}$ with $I_j = (c_{j-1}+1) : c_j$. From (2) the log-likelihood function is given by:

$$l(\mathrm{C}; \mathbf{x}^n) = \sum_{i=1}^n \sum_{j=1}^{|\mathrm{C}|} \log f_{\hat{\theta}_j}(x_{(c_{j-1}+1):c_j}^{(i)}). \tag{3}$$

Let $R : \mathcal{C} \to \mathbb{R}$ denote some regularization function and $J : \mathbb{N} \to \mathbb{R}$ an increasing function on the sample size n. We introduce the penalized likelihood estimator based on the functions R and J in the following definition.

Definition 1. *Given a sample* $\mathbf{x}^n$ *and a constant* $\lambda > 0$, *the Penalized Likelihood (PL) function for the set of change-points* C *is defined as:*

$$\mathrm{PL}(\mathrm{C};\mathbf{x}^n) = -l(\mathrm{C};\mathbf{x}^n) + \lambda R(\mathrm{C})J(n). \tag{4}$$

The PL estimator for the change-point set is then defined as:

$$\hat{\mathrm{C}}(\mathbf{x}^n) = \operatorname*{argmin}_{\mathrm{C}\in\mathcal{C}} \mathrm{PL}(\mathrm{C};\mathbf{x}^n). \tag{5}$$

As we will show later in Theorems 1 and 2, in order to obtain the consistency of the change-point estimator defined by (5), we need the functions R and J to satisfy some properties. This will be made precise in the statements of these theorems.

3 Algorithms

In order to efficiently estimate the change-point set, we suppose in this section that the regularization function R is additive. That means that there exists a function $\rho : \{1, \ldots, m\}^2 \to \mathbb{R}$ such that for $C = \{c_0, \ldots, c_k\}$ we have:

$$R(\mathrm{C}) = \sum_{j=1}^{k_{\mathrm{C}}} \rho(c_{j-1}+1, c_j). \tag{6}$$

3.1 Dynamic programming segmentation algorithm

As presented in Jackson et al. (2005), dynamic programming can be used to calculate exactly the PL estimator. Under an additive regularization, the function we want to minimize can be written as:

$$-l(\mathrm{C};\mathbf{x}^n) + \lambda J(n)R(\mathrm{C}) = \sum_{j=1}^{k_{\mathrm{C}}} Q((c_{j-1}+1):c_j),$$

where

$$Q((c_{j-1}+1):c_j) = -\log f_{\hat{\theta}_j}\left(x^{(i)}_{(c_{j-1}+1):c_j}\right) + \lambda J(n)\rho(c_{j-1}+1, c_j).$$

The equation shows that we can completely decouple the loss from different blocks. Let $\mathcal{C}_i$ be the set of all ordered change-point sets in $1:i$. Define

$$F(i) = \min_{\mathrm{C}\in\mathcal{C}_i}\left\{\sum_{j=1}^{k} Q((c_{j-1}+1):c_j)\right\}$$

as the optimal value for the segmentation up to variable i. The estimator $\hat{\mathrm{C}}(\mathbf{x}^n)$ given by (5) is obtained by computing $F(m)$. But notice that

$$F(i) = \min_{c\in(k-1):(i-1)} \{F(c) + Q((c+1):i)\},$$

which establishes a recursion equation for the values of $F(i)$, i varying from 1 to m. The value of $F(1)$ can be computed trivially, and then we use the recursion to compute the values until we reach $F(m)$.

Albeit m is fixed and only n grows, the number of variables m can be very large in some applications, so it is useful to express the complexity in terms of both. The dynamic programming segmentation algorithm runs on a time complexity of $O(m^2)$. However, we are assuming that Q have been previously computed for all intervals in $1:m$. To compute Q, we need to compute the maximum likelihood estimators for each block. For most models, this can be done efficiently by computing the sufficient statistics, and then compute the entries of Q. In the case where no fixed dimension sufficient statistics exist, we need to reprocess the data every time, so the complexity to compute Q is $O(nm^3)$. Hence, the final time complexity of the algorithm is $O(T(n,m) + m^2)$. In the worst case the algorithm is $O(nm^3)$, and can be very slow for big values of m.

Depending on the function ρ chosen, the PELT algorithm of Killick et al. (2012) might be applicable. It consists of an adaptation of the dynamic programming algorithm discussed here. In some scenarios, such as when the number of change-points is proportional to the number of variables, it runs in $O(m)$. The final complexity would be $O(nm + m)$ when suitable sufficient statistics exist.

3.2 Hierarchical segmentation algorithm

For efficient computation of $\hat{\mathrm{C}}(\mathbf{x}^n)$ we can use an approximation to the optimum in (5), known as hierarchical segmentation or binary segmentation, first proposed in Scott and Knott (1974). Given an integer interval $I = r:s$, write $\mathbf{x}^n_I$ for the data with columns restricted to I and assume the penalizing function R is additive. Define the penalized loss of I as:

$$\begin{aligned}\mathrm{PL}(I) &= \mathrm{PL}(\{0, s-r+1\};\mathbf{x}^n_I)\\ &= -l(\{0, s-r+1\};\mathbf{x}^n_I) + \lambda\rho(r,s)J(n),\end{aligned}$$

with the appropriate renumbering of the columns in $\mathbf{x}^n_I$. That is, the penalized loss corresponding to the interval I is the penalized loss defined in (4) when we only consider the data $\mathbf{x}^n_I$ and perform no splits. We use the convention $\mathrm{PL}(\varnothing) = 0$. For $c \in I = r:s$, define:

$$h_I(c) = \mathrm{PL}(r:c) + \mathrm{PL}((c+1):s). \tag{7}$$

Observe that when $c = s$, by convention, we have $(s+1):s = \varnothing$ so that $h_I(s) = \mathrm{PL}(I)$.

The hierarchical segmentation algorithm works recursively as follows. It begins with the set $\hat{C}_{bs}(\mathbf{x}^n) = \{0, m\}$ corresponding to the single interval $I = 1:m$. In each iteration and for each interval I determined by $\hat{C}_{bs}(\mathbf{x}^n)$, the algorithm computes $\hat{c} = \mathrm{argmin}_{c\in I}\, h_I(c)$ and adds it to $\hat{C}_{bs}(\mathbf{x}^n)$. Observe that if in one iteration $\hat{c} = s$, as $s \in \hat{C}_{bs}(\mathbf{x}^n)$, no changes are made on $\hat{C}_{bs}(\mathbf{x}^n)$. The process continues until no more points can be added to $\hat{C}_{bs}(\mathbf{x}^n)$.

To evaluate PL at each interval, we either store all possible values in the same fashion as for the dynamic programming algorithm or we evaluate them on the run. Since storing would make the algorithm $O(m^2)$ in any scenario, it is more interesting to precompute the sufficient statistics and evaluate the loss on the intervals as they appear.

In the worst-case scenario, the algorithm has time complexity of order $O(T(n,m) + m^2)$. However, the algorithm has asymptotic complexity of order $O(T(n,m) + mk_{C^*}))$, as proved in the Supplementary Material.

3.3 Consistency of the algorithms

In this section, we state the theoretical results that guarantee the consistency of the estimator (5) computed exactly by the dynamic programming algorithm or approximated by the hierarchical segmentation algorithm. For each method, we state a different set of assumptions that must be satisfied by the family of probability distributions considered in the model.

Assumption 1. *Suppose there exists a function* $l^* : \mathcal{C} \to \mathbb{R}$ *such that*

(PL1) For any $\mathrm{C} \in \mathcal{C}$ *we have that* $\frac{1}{n}l(\mathrm{C};\mathbf{x}^n) \to l^*(\mathrm{C})$ *almost surely as* $n \to \infty$, *where* l *is the log-likelihood function defined in (3). Moreover, assume there exists* $\alpha > 0$ *such that*

$$\inf_{\mathrm{C}\not\supseteq \mathrm{C}^*} l^*(\mathrm{C}) \le l^*(\mathrm{C}^*) - \alpha < l^*(\mathrm{C}^*).$$

(PL2) There exists a sequence $\{v(n)\}_{n\in\mathbb{N}}$ *satisfying* $v(n) \to \infty$ *and* $v(n)/n \to 0$ *when* $n \to \infty$, *and such that for any* $\mathrm{C} \supseteq \mathrm{C}^*$ *we have that* $|l(\mathrm{C};\mathbf{x}^n) - l(\mathrm{C}^*;\mathbf{x}^n)| < v(n)$ *eventually almost surely as* $n \to \infty$.

Observe that (PL2) implies that $l^*(\mathrm{C}) = l^*(\mathrm{C}^*)$ for all $\mathrm{C} \supseteq \mathrm{C}^*$.

We now state the consistency result of the PL estimator given in (5).

Theorem 1. *Suppose that the family* $\mathcal{F}$ *of probability distributions satisfy Assumption 1. Let* R *be a penalizing function such that* $R(\mathrm{C}) > R(\mathrm{C}')$ *whenever* $\mathrm{C} \supset \mathrm{C}'$ *and let* $J(n)$ *be such that*

$J(n)/v(n) \to \infty$ and $J(n)/n \to 0$ when $n \to \infty$. Then the estimator of the change-point set given by (5) satisfies $\hat{C}(\mathbf{x}^n) = C^*$ eventually almost surely as $n \to \infty$.

Notice that the regularization function R does not need to be additive to guarantee the consistency of the PL estimator. However, this is a desirable property to efficiently compute the estimator by using the dynamic programming segmentation algorithm.

In order to prove that the estimator given by the hierarchical segmentation algorithm is also consistent, we need a slightly different set of hypotheses considering the local nature of this algorithm. Denote by $\mathcal{I}$ the set of all intervals $I \subset 1:m$. Given $I \in \mathcal{I}$, denote by $\hat{\theta}_I$ the maximum likelihood estimator of the parameter θ on the interval I and as before, let $\mathbf{x}_I^n$ be the data restricted to the interval I. Define the maximum log-likelihood function for the interval I as:

$$l(I;\mathbf{x}_I^n) = \sum_{i=1}^{n} \log f_{\hat{\theta}_I}(x_I^{(i)}).$$

Let h_I be the loss function for the interval I as defined in (7).

Assumption 2. *There exists a function $l^* : \mathcal{I} \to \mathbb{R}$ such that*

(H1) For any integer interval $I \in \mathcal{I}$ we have that $\frac{1}{n} l(I;\mathbf{x}_I^n) \to l^*(I)$ almost surely as $n \to \infty$. If $I = r:s$, defining $h_I^* : I \to \mathbb{R}$ as:

$$h_I^*(u) = -l^*(r:u) - l^*((u+1):s), \quad u \in I, \tag{8}$$

we have that $\min_{c \in I \setminus \{s\} \cap C^*} h_I^*(c) < \min_{c \notin I \setminus \{s\} \cap C^*} h_I^*(c)$.

(H2) There exists a sequence $\{v(n)\}_{n \in \mathbb{N}}$ satisfying $v(n) \to \infty$ and $v(n)/n \to 0$ when $n \to \infty$, and such that, for any integer interval $I = r:s$ satisfying $I \setminus \{s\} \cap C^* = \emptyset$ we have

$$\max_{u \in I} |l(I;\mathbf{x}_I^n) - l(r:u;\mathbf{x}_I^n) - l((u+1):s;\mathbf{x}_I^n)| < v(n)$$

eventually almost surely as $n \to \infty$.

We can now state the consistency of the change-point set estimator given by the hierarchical segmentation algorithm.

Theorem 2. *For any $\lambda > 0$, let $\hat{C}_{hs}(\mathbf{x}^n)$ be the estimator computed by the hierarchical segmentation algorithm. Suppose that the family $\mathcal{F}$ of probability distributions satisfy Assumption 2. Suppose that R satisfies (6) for some function $\rho : \mathcal{I} \to \mathbb{R}$ and that $\rho(I) < \rho(I_1) + \rho(I_2)$ whenever $I = I_1 \cup I_2$, with $I_1, I_2 \neq \emptyset$. Finally, assume that the function $J(n)$ satisfies $J(n)/v(n) \to \infty$ and $J(n)/n \to 0$ when $n \to \infty$. Then, $\hat{C}_{hs}(\mathbf{x}^n) = C^*$ eventually almost surely as $n \to \infty$.*

The proof of Theorems 1 and 2 are included in the Supplementary Material to this article, available online. There, we also show that the family of Bernoulli and Gaussian random variables satisfy Assumptions 1 and 2. Hence, both the dynamic programming and hierarchical segmentation algorithms provide consistent estimators of the change-point parameters (C^*, θ^*) for these families of distributions.

3.4 FRV methodology for penalizing constant selection

Albeit the algorithms are consistent for any value of the penalization constant λ, their performance, in practice, can drastically improve if we choose good values of the constant. To this end, a reference methodology was proposed by Lavielle (2005) and efficiently computed in the CROPS algorithm by Haynes et al. (2017). The idea consists of plotting the cost function, the negative log-likelihood, versus the number of change-points and selecting the value at which the curve starts to become flat, a technique similar to the elbow plot used in cluster analysis. This approach is widely used in practice; however, it has some drawbacks. For instance, it requires visual inspection, hence human intervention, making it nonautomatic. Because of this, one cannot study the consistency of the whole procedure. Moreover, the correct choice of the number of change-points is difficult due to the cost function's lack of a reference scale.

To address these issues, we propose a similar idea to that of the elbow plot, but on different variables. Instead of plotting the cost function versus the number of change-points, we see how the proportion of change-points varies against the values of the penalization constant. Chosen $\gamma > 0$ and $\lambda_{\max} > 0$, the idea is to evaluate the detected number of change-points for each constant value in the grid $\{0, \gamma, 2\gamma \ldots, b\gamma\}$, where $b = \lfloor \frac{\lambda_{\max}}{\gamma} \rfloor$.

In order to simplify the notation, let $\hat{l}(k)$ be the log-likelihood of the best split with k change-points evaluated at the MLE and $R(k)$ be the regularization of the best change-point set. Fixed λ, consider the function $PL(k) = -\hat{l}(k) + \lambda J(n) R(k)$ as the Penalized Likelihood for the best model with k change-points. If, for every λ, the function is strictly convex, then the model estimates $\hat{k} = k$ if, and only if, both equations below hold:

$$\begin{aligned} -\hat{l}(k) + \lambda J(n) R(k) &\leq -\hat{l}(k+1) + \lambda J(n) R(k+1) \\ -\hat{l}(k) + \lambda J(n) R(k) &< -\hat{l}(k-1) + \lambda J(n) R(k-1). \end{aligned}$$

For each k, define the interval

$$A(k) = \left[\frac{\hat{l}(k+1) - \hat{l}(k)}{J(n)(R(k+1) - R(k))}, \frac{\hat{l}(k) - \hat{l}(k-1)}{J(n)(R(k) - R(k-1))} \right),$$

making the convention that this interval is empty if the left extreme is greater than the right one. The method selects $\hat{k} = k$ if, and only if, we have $\lambda \in A(k)$.

Under the assumptions of Theorem 1 or Theorem 2, we have $|A(k)| \in O\left(\frac{v(n)}{J(n)}\right) \overset{a.s.}{\to} 0$ if $k > k^*$, $A(k^*) \overset{a.s.}{\to} (0, +\infty 0$ and $|A(k^*)| \in O\left(\frac{n}{J(n)}\right) \overset{a.s.}{\to} +\infty$, where $|A(k)|$ denote the length of the interval $A(k)$. The behavior for $A(k)$ for $k < k^*$ is unimportant for the procedure.

Choose the step size $\gamma = \gamma(n)$ such that $\frac{v(n)}{J(n)} \ll \gamma(n)$ and $\gamma(n) \to 0$ as $n \to \infty$. Taking a sample size that is large enough, if at the grid point $i\gamma$ the algorithm outputs $k > k^*$, then at the point $(i+1)\gamma$ it will output k' such that blue$k > k' \geq k^*$. This is due to the fact that the step size is greater than the interval length $|A(k)|$. If the algorithm outputs k^*, then at the next point it will output k^* again, now due to γ being smaller than $|A(k^*)|$. Therefore, the key idea is to fit the models until we get a repeated value for the estimated number of change-points. If no repeated values occur in the given interval, we can just halve γ and evaluate in the new grid until a repeated value appears.

The First Repeated Value (FRV) procedure outputs the penalization value and model such that the first repeated value of the number of change-points is detected. The pseudocode is given below. Assume we have a procedure fitModel that fits the model for a given penalization constant and getNCP that extracts the number of

Algorithm 1 First Repeated Value procedure

```
procedure FRV(X, $\gamma$, $\lambda_{\max}$, m)
  while True do
    lastNCP ← 1        ▷ at $\lambda = 0$ the proportion is always 1
    $b \leftarrow \lfloor \frac{\lambda_{\max}}{\gamma} \rfloor$
    for $i \in \{1, \ldots, b\}$ do
      model ← $fitModel(X, i\gamma)$
      currNCP ← $getNCP(model)/m$
      if lastNCP = currNCP then
        return $(model, currNCP)$        ▷ output the model
      end if
      lastNCP ← currNCP
    end for
    $\gamma \leftarrow \gamma/2$        ▷ halves $\gamma$ since no repetition was found
  end while
end procedure
```

change-points.To exemplify how $\gamma(n)$ can be chosen, take $J(n) = \log(n)$ and consider the families of Normal distributions with unknown mean and variance. In the Appendix and in the Supplementary Material, we show that $v(n) = O(\log(\log(n))$. Then we can take $\gamma(n) = (\log(n))^{-1/2}$ and we have that $\frac{v(n)}{J(n)} \ll \gamma(n)$, with $\gamma(n) \to 0$ as $n \to \infty$.

We now state a theorem on the consistency of the FRV procedure. The proof is presented in the Supplementary Material.

Theorem 3. *Assume that the model satisfies the consistency conditions for either Theorem 1 or Theorem 2. Let* $\lambda_{\max} > 0$ *be constant and assume* $\frac{v(n)}{J(n)} \ll \gamma(n)$ *and assume that, for every constant* λ, *the Penalized Likelihood* $PL(k) = -\hat{l}(k) + \lambda J(n)R(k)$ *is strictly convex. Let* $\hat{C}_{FRV}$ *be the change-point set estimated by the FRV procedure. Then, we have* $\hat{C}_{FRV} = C^*$ *eventually almost surely as* $n \to \infty$.

This approach has the stated properties; first, the proportion of change-points always varies from 1 to 0, so that it is scale free from the statistical family, sample size, and number of change-points; second, it allows us increasingly refine the penalization grid; third, it allows for an automatic constant selection method that is consistent.

4 Implementation

The exact dynamic programming algorithm and the hierarchical segmentation algorithm described in Section 3 were implemented using the R software R Core Team (2020) in the R package `blockcpd`. The performance of both algorithms was compared to the classical CUSUM method for sequence segmentation on model-based simulated data. As the hierarchical algorithm showed similar performance to the exact approach and is computationally more efficient, it was applied to a real SNPs dataset to detect ROH islands. The results were confronted against the ROH islands detected by PLINK and by Pemberton et al. (2012) (results available in the Supplementary Material).

4.1 Simulations

In the model-based simulations, we consider two families of distributions: the Bernoulli distribution and the Gaussian distribution, with both mean and variance unknown (results available in the Supplementary Material). We fixed the number of variables m and varied the number of change-points in the model.

The simulated datasets were generated as follows. For each probability distribution (Bernoulli and Normal), we varied the number of samples n from 50 to 500 in steps of 50. For each sample size, we simulated 1000 datasets with n samples. We fixed the number of variables as $m = 200$ and the number of change-points $|C^*| = k^*$ to take values in $\{10, 50\}$. For each k^*, the change-points were sampled without replacement from a uniform distribution in $[1, 199]$ and where maintained fixed for all datasets and all sample sizes. Bernoulli parameters were sampled independently from a uniform distribution. For the Gaussian distribution, means were sampled independently from a $N(0, \sqrt{5})$, and variances from an Exp(1). These parameters also remained fixed for all datasets and all sample sizes.

For each dataset, we computed the change-points with both the dynamic programming segmentation algorithm and the hierarchical segmentation algorithm, as described in Section 3. In order to select the penalization constant, we applied the FRV methodology for the interval $[0, 10]$ with steps $\gamma = \frac{1}{\sqrt{\log(n)}}$.

In order to compare our methods with the traditional approach CUSUM, we applied the `changepoint` package from Killick and Eckley (2014) on the series consisting of the average of each variable (column). We used a mean-variance change-point detection algorithm assuming normality, considering the PELT search algorithm and the BIC as a regularization.

To evaluate the convergence of the change-point set estimated by the algorithms, we considered the Jaccard Index, defined by $J(C_1, C_2) = \frac{|C_1 \cap C_2|}{|C_1 \cup C_2|}$. We also took into account the convergence of the estimated number of change-points to k^*. It allow us to see when the model is heavily underestimating or overestimating the number of change-points.

Figure 1 shows the comparison of the performance of the algorithms as the sample sizes grow. The boxplots median is approximating the true number of change-points and 1 for the Jaccard Index. Moreover, they are becoming increasingly narrower and closer to 1, which indicates total similarity between the change-point sets. The average lines are also shown to help us envisage the behavior of the algorithms as sample size increases.

For the Bernoulli and Normal simulation scenarios, the approximate algorithm has very similar performance to the exact dynamic programming algorithm. Indeed, their boxplots are almost indistinguishable. For small sample size, as expected, the hierarchical algorithm exhibits greater variance.

In the Bernoulli scenario, the CUSUM method performs much worse than our algorithms, as indicated by Fig. 1. This is expected since our statistical model is correct for the data in this case, and the CUSUM method relies on the normal approximation to detect changes in mean and variance. For the Normal scenario, when $k^* = 10$, the CUSUM method is very competitive and shows smaller variance than our algorithms. For $k^* = 50$, our algorithms outperform again the CUSUM method.

It is also important to mention that the FRV methodology plays a key role in selecting the constant in the simulation. Indeed, it would be a burdensome task to perform visual inspection for each dataset. The models are being fit in an increasing grid of penalization constants. As shown by the simulation results, the FRV is automatically selecting models with increasingly better performance as the sample size grows.

4.2 ROH islands on African and European populations

As described in the introduction, we propose to frame the problem of ROH islands detection as a change-point detection problem assuming a Bernoulli marginal distribution for each codified SNP. Observe that we do not need to assume independence between different SNPs in order to have consistent estimators of the change-points, we only assume consecutive parameters on the blocks of the distribution to be different. In particular, the ROH islands can be determined as those blocks with a high value of the estimated parameter.

We can use domain knowledge to construct a proper regularization function. The first consideration is that the distance between SNPs is not uniform. That is, the distance between the i-th and j-th SNP is not $|i - j|$, but rather $|B(i) - B(j)|$, where B is a function that maps each SNP to its physical location on the chromosome. The physical location of a SNP is measured as the number of base pairs before that particular SNP. The second observation is that very small blocks are usually not interesting for the analyst. It is usual to set a minimum block size in which SNPs are grouped.

Considering these observations, we define the regularization function ρ for the block $r : s$ as:

$$\rho(r, s) = \begin{cases} +\infty & \text{if } \dfrac{|B(s) - B(r)|}{\beta} \leq T \\ \dfrac{1}{|B(s) - B(r)|/\beta} & \text{otherwise.} \end{cases}$$

In the expression above, T denotes a threshold for the minimal physical distance allowed for an ROH island, and $\beta = 10^6$ is a scaling factor to work on mega bases unit. The regularization function $R(C)$ in (4) is then defined as the sum of the function ρ over the different blocks in C, as in (6).

The SNPs data we analysed was obtained from the Human Genome Diversity Project (HGDP), and consists of approximately 600 000 SNP markers from Illumina HuHap 650k platform (Li et al. 2008). We considered individuals from African and European populations. On this dataset, each row represents an

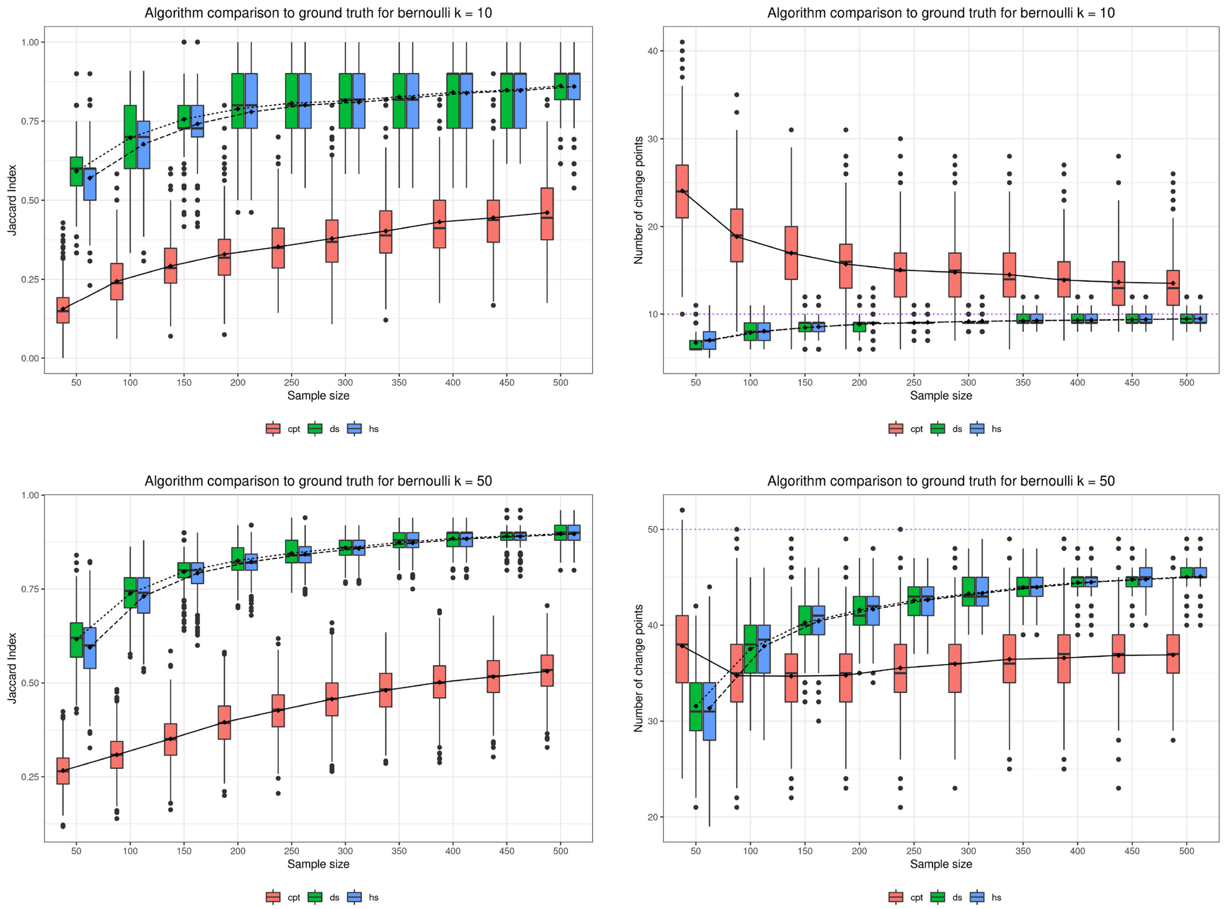

Figure 1. Comparison of estimated change-point sets between CUSUM (cpt), dynamic programming segmentation (ds), and hierarchical segmentation (hs) algorithms for the Bernoulli family using the Jaccard Index (left) and the number of change-points (right). The true number of change-points k is indicated by the dotted horizontal line

individual from the population, and each column corresponds to each SNP.

We estimated the change-point set and parameters on each block for each population using a value of λ that was selected using the FRV in the interval $[0, 10]$ with steps of $\gamma = \frac{1}{\sqrt{\log(n)}}$. The threshold size T was set to be 1% of the chromosome size, and the penalization for the sample size set to $J(n) = \sqrt{n}$.

For each population, we also performed ROH identification for each individual with the criteria described by McQuillan et al. (2008) and Kirin et al. (2010), using the software PLINK v1.9 (Purcell et al. 2007).

Figure 2 shows the ROH islands, for African and European populations, detected across the genome by considering a quantile cutoff of 95%. For PLINK, the cutoff is based on the frequency distribution that SNPs appear in ROH. For `blockcpd`, the cutoff is based on the probability parameter of the blocks. We see an overlap between the estimates for almost all chromosomes. The intersection percentage is shown on the vertical axis. We observe that PLINK tends to subdivide regions, while our method provided more contiguous estimates, as was the goal when we introduced the regularization function and the threshold parameter T.

A comparison of the ROH islands estimates for a quantile cutoff of 99% is provided in the Supplementary Material, where we also included the analysis of full genome sequences from the HGDP database and the comparison of our method with the method by Pemberton et al. (2012).

5 Discussion

In this paper, we proposed a new change-point detection method based on penalized maximum likelihood and proved its consistency, for two different algorithms. This new approach is motivated by the problem of identifying homozygosity islands on the genome of individuals in a population. Our method directly tackles the issue of determining the homozygosity islands at the population level without analysing single individuals and then combining the results, as is made nowadays in state-of-the-art approaches. Applying this method to real data of two populations from the Human Genome Diversity Project (HGDP) showed the potentiality of these algorithms to highlight highly homozygous regions in the genome. The nonhomogeneity of the regularization function R can provide flexibility to incorporate more domain knowledge of the application area.

We also propose the FRV methodology to automatically and efficiently select the penalization constant λ. We proved that the methodology preserves the consistency property even if fitting the model in an increasing grid of penalization constants. The simulation study shows that the methodology chooses models with overall good performance.

From the theoretical point of view there is much to explore in future research. A first step will be to check if the assumptions implying the consistency results hold for a wider class of distributions, such as the Exponential Family, finite-state Markov Chains and Multivariate Gaussian distributions within each block. A second interesting question is to prove or give a counter-example of whether

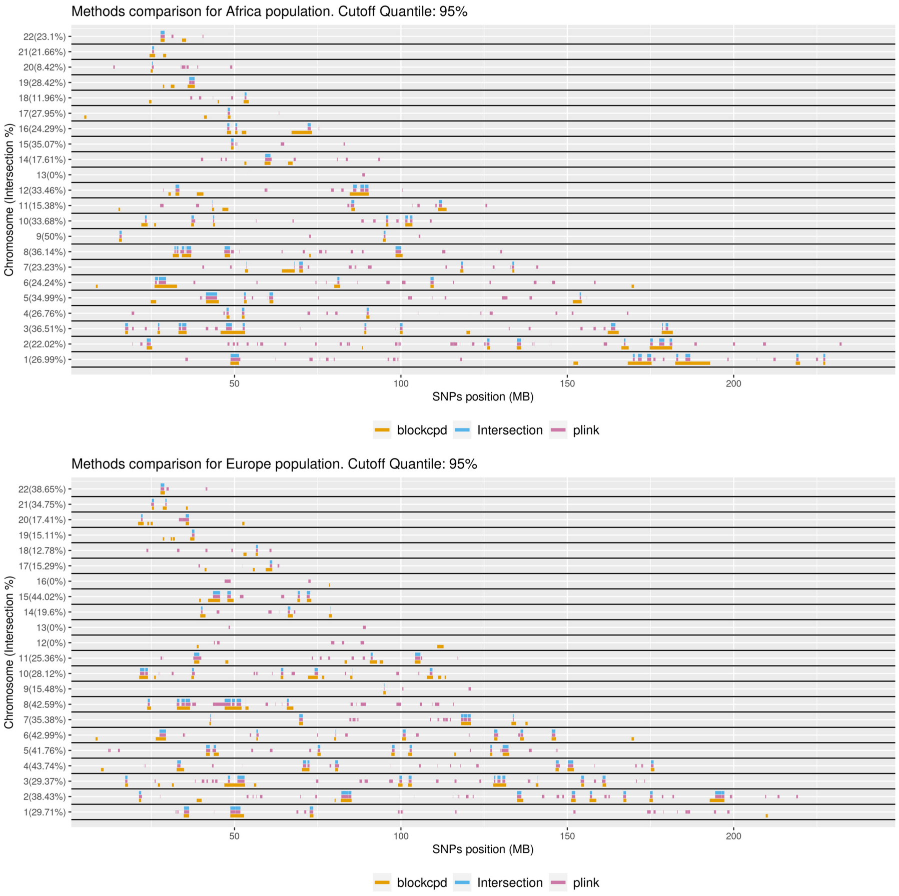

Figure 2. Comparison of the detected ROH islands across the genome by our hierarchical algorithm (blockcpd) and the PLINK method. Both ROH islands detected by PLINK and our method are shown, together with their intersection. The intersection percentage is displayed on the vertical axis

Assumption 2 implies Assumption 1; that is, if the convergence of the greedy algorithm implies the convergence of the exact algorithm as well.

Conflict of interest: None declared.

Funding

This work was supported by the São Paulo Research Foundation, Brazil [2013/07699-0, 17/10555-0, 20/10136-0, 21/06860-8]; Coordination of Superior Level Staff Improvement, Brazil [Ph.D. fellowship to L.P.]; and the National Council for Scientific and Technological Development, Brazil, [311763/2020-0 to F.L.].

Supplementary data

Supplementary data is available at *Bioinformatics* online.

Data availability

The data underlying this article are available in the Human Genome Diversity Project (HGDP) at https://www.science.org/doi/10.1126/science.aay5012. The datasets were derived from sources in the public domain: ftp://ngs.sanger.ac.uk/production/hgdp/hgdp_wgs.20190516/hgdp_wgs.20190516.full.chr[i].vcf.gz (i = 1 to 22).

References

Agudelo-España D, Gomez-Gonzalez S, Bauer S *et al.* Bayesian online prediction of change points. In: *Proceedings of the 36th International Conference on Uncertainty in Artificial Intelligence (UAI)*, Vol. 124. Proceedings of Machine Learning Research, 2020, 320–329.

Castro BM, Lemes RB, Cesar J *et al.* A model selection approach for multiple sequence segmentation and dimensionality reduction. *J Multivar Anal* 2018; **167**:319–30.

Ceballos FC, Joshi PK, Clark DW *et al.* Runs of homozygosity: windows into population history and trait architecture. *Nat Rev Genet* 2018;**19**: 220–34.

Chen J, Gupta A. *Parametric Statistical Change Point Analysis*. 2nd edn. Boston: Birkhäuser Boston, 2012.

Chen J, Gupta AK. Testing and locating variance changepoints with application to stock prices. *J Am Stat Assoc* 1997;**92**:739–47.

Chow GC. Tests of equality between sets of coefficients in two linear regressions. *Econometrica* 1960;**28**:591.

Fryzlewicz P. Wild binary segmentation for multiple change-point detection. *Ann Statist* 2014;**42**:2243–81.

Hawkins D, Zamba K. A change-point model for a shift in variance. *J Qual Technol* 2005;**37**:21–31.

Haynes K, Eckley IA, Fearnhead P. Computationally efficient changepoint detection for a range of penalties. *J Comput Graph Stat* 2017;**26**:134–43.

Hinkley D. Inference about the change-point in a sequence of random variables. *Biometrika* 1970;**57**:1–17.

Jackson B, Scargle J, Barnes D *et al.* An algorithm for optimal partitioning of data on an interval. *IEEE Signal Process Lett* 2005;**12**:105–8.

Keshavarz H, Scott C, Nguyen X. Optimal change point detection in Gaussian processes. *J Stat Plann Inference* 2018;**193**:151–78.

Killick R, Eckley IA. Changepoint: an R package for changepoint analysis. *J Stat Soft* 2014;**58**:1–19.

Killick R, Fearnhead P, Eckley IA. Optimal detection of changepoints with a linear computational cost. *J Am Stat Assoc* 2012;**107**:1590–8.

Kirin M, McQuillan R, Franklin CS *et al.* Genomic runs of homozygosity record population history and consanguinity. *PLoS One* 2010;**5**:e13996.

Lavielle M. Using penalized contrasts for the change-point problem. *Signal Process* 2005;**85**:1501–10.

Lee T-S. Change-point problems: bibliography and review. *J Stat Theory Pract* 2010;**4**:643–62.

Lemes RB, Nunes K, Carnavalli JE *et al.* Inbreeding estimates in human populations: applying new approaches to an admixed Brazilian isolate. *PLoS One* 2018;**13**:e0196360.

Leonardi F, Lopez-Rosenfeld M, Rodriguez D *et al.* Independent block identification in multivariate time series. *J Time Ser Anal* 2021;**42**:19–33.

Leutenegger A-L, Sahbatou M, Gazal S *et al.* Consanguinity around the world: what do the genomic data of the HGDP-CEPH diversity panel tell us? *Eur J Hum Genet* 2011;**19**:583–7.

Li G, Wang J, Liang J *et al.* The application of a double CUSUM algorithm in industrial data stream anomaly detection. *Symmetry* 2018;**10**:264.

Li JZ, Absher DM, Tang H *et al.* Worldwide human relationships inferred from genome-wide patterns of variation. *Science* 2008;**319**:1100–4.

Matteson DS, James NA. A nonparametric approach for multiple change point analysis of multivariate data. *J Am Stat Assoc* 2014;**109**:334–45.

McQuillan R, Leutenegger A-L, Abdel-Rahman R *et al.* Runs of homozygosity in European populations. *Am J Hum Genet* 2008;**83**:359–72.

Niu YS, Hao N, Zhang H. Multiple change-point detection: a selective overview. *Stat Sci* 2016;**31**:611–23.

Page ES. Continuous inspection schemes. *Biometrika* 1954;**41**:100–15.

Pemberton TJ, Absher D, Feldman MW *et al.* Genomic patterns of homozygosity in worldwide human populations. *Am J Hum Genet* 2012;**91**:275–92.

Purcell S, Neale B, Todd-Brown K *et al.* PLINK: a tool set for whole-genome association and population-based linkage analyses. *Am J Hum Genet* 2007;**81**:559–75.

Qu Z, Perron P. Estimating and testing structural changes in multivariate regressions. *Econometrica* 2007;**75**:459–502.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing, 2020.

Rice G, Zhang C. Consistency of binary segmentation for multiple change-point estimation with functional data. *Stat Probab Lett* 2022;**180**:109228.

Scott A, Knott M. A cluster analysis method for grouping means in the analysis of variance. *Biometrics* 1974;**30**:507–12.

Tartakovsky AG, Rozovskii BL, Blazek RB *et al.* A novel approach to detection of intrusions in computer networks via adaptive sequential and batch-sequential change-point detection methods. *IEEE Trans Signal Process* 2006;**54**:3372–82.

Truong C, Oudre L, Vayatis N. Selective review of offline change point detection methods. *Signal Process* 2020;**167**:107299.

Tsay RS. Outliers, level shifts, and variance changes in time series. *J Forecast* 1988;**7**:1–20.

Venkatraman ES. Consistency results in multiple change-point problems. Ph.D. Thesis, Stanford University, 1992.