




Efficient sparsity adaptive changepoint estimation

Per August Jarval Moen¹ ,
Ingrid Kristine Glad¹ , and Martin Tveten² 

¹*Department of Mathematics, University of Oslo*
e-mail: pamoen@math.uio.no; glad@math.uio.no

²*Norwegian Computing Center*
e-mail: tveten@nr.no

Abstract: We propose a computationally efficient and sparsity adaptive procedure for estimating changes in unknown subsets of a high-dimensional data sequence. Assuming the data sequence is Gaussian, we prove that the new method successfully estimates the number and locations of changepoints with a given error rate and under minimal conditions for all sparsities of the changing subset. Our method has computational complexity linear up to logarithmic factors in both the length and number of time series, making it applicable to large data sets. Through extensive numerical studies we show that the new methodology is highly competitive in terms of both estimation accuracy and computational cost. The practical usefulness of the method is illustrated by analysing sensor data from a hydro power plant, and an efficient R implementation is available.

MSC2020 subject classifications: Primary 62H12.

Keywords and phrases: Changepoints, high-dimensional, binary segmentation.

Received June 2024.

1. Introduction

During the last decades, new technology has made it possible to gather data in larger quantities from an ever wider range of sources. Data can often display non-stationarities in the form of distributional changes over time, leading to incorrect statistical inferences if not accounted for. Inference on changepoints may also be of interest in itself. For instance, Cunen, Hjort and Nygård [6] search for changes in the number of battle deaths in interstate wars between 1816 and 2007, Gao et al. [8] study monitoring of the temperature of transplant organs, and Tveten, Eckley and Fearnhead [20] use a changepoint detection algorithm for condition monitoring of a subsea pump.

In this paper, we study the problem of detecting and estimating an unknown number of changes in the mean vector of high-dimensional data. By *detection*, we refer to testing for the presence of one or more changepoints in the data. By *estimation*, we refer to estimation of the location(s) of the detected changepoint(s). This problem is well understood in the literature for univariate data.

Several computationally efficient algorithms have been proposed during the last decade, including Pruned Exact Linear Time of Killick, Fearnhead and Eckley [11], Wild Binary Segmentation of Fryzlewicz [7], Narrowest Over Threshold of Baranowski, Chen and Fryzlewicz [1] and Seeded Binary Segmentation of Kovács et al. [12]. Notably, these methods have been shown to achieve near optimal performance, in a minimax sense, see Wang, Yu and Rinaldo [22].

Several methods for the multivariate change in mean problem have also been proposed, although this problem is somewhat less studied than the univariate setting. The Inspect method of Wang and Samworth [21] uses sparse projections of CUSUM statistics and a variant of Wild Binary Segmentation to detect and localize multiple sparse changes in the mean. Cho and Fryzlewicz [4] propose the Sparsified Binary Segmentation algorithm based on thresholding and aggregating CUSUM statistics over coordinates, in combination with Binary Segmentation. The Double CUSUM method of Cho [3] uses test statistics based on ordered CUSUMs, in combination with ordinary Binary Segmentation. The SUBSET method of Tickle, Eckley and Fearnhead [19] uses a penalized likelihood approach, in combination with the Wild Binary Segmentation search procedure, while the methods of Kaul et al. [10] and Kaul and Michailidis [9] use a locally refitted least squares estimator.

In this work, we present a novel multiple changepoint estimation algorithm, which we call ESAC (**E**fficient **S**parsity **A**daptive **C**hangepoint estimator). The method is designed to detect and estimate the locations of an unknown number of changes in the mean of high-dimensional data sequences. An important feature of ESAC is that the subset of data components that undergo a change need not be known — it can be anything from a single changing component to a small subset to all components. We refer to the size of the changing subset as the *sparsity* of the change. ESAC comes with strong theoretical guarantees, and is in particular adaptive to all sparsities of changes and all distances between changepoints, both of which are allowed to vary among the changepoints. Still, the worst-case computational cost of ESAC is linear in the number of observations, n , as well as the number of components, p , save for logarithmic factors. Via simulations, we demonstrate that ESAC is highly competitive in terms of statistical accuracy and run time.

We summarize the novelty of our work in the following:

1. We demonstrate how the single changepoint testing procedure of Liu, Gao and Samworth [13], which is based on hard-thresholding, can be modified for multiple changepoint search. This modification is non-trivial, as both thresholding levels and tuning parameters need to be adjusted appropriately to allow for control over the Type I error.
2. Based on the modified changepoint testing procedure, we construct a novel estimator of the location of a single changepoint and prove that it has strong theoretical properties.
3. By combining our proposed test statistic and changepoint estimator, we propose a novel method, ESAC, for multiple changepoint detection and estimation. The method uses a variant of Seeded Binary Segmentation

[12] and Narrowest-over-Threshold [1] for selection of changepoints. The new method comes with strong statistical and computational theoretical guarantees, and enjoys a certain kind of optimality.

4. We carry out an extensive simulation study, where we compare our single- and multiple changepoint estimators to several state-of-the-art methods. The simulation study evaluates the performance of the competing methods both with a single and multiple changepoints. We also investigate the performance of the methods under misspecified models, such as heavy-tailed noise and temporal and spatial auto-correlation.
5. To illustrate the applicability of ESAC, we apply the method to analyze sensor data from a Swedish hydro power plant.

ESAC is implemented in an R package **HDCD** [14], available on The Comprehensive R Archive Network (cran.r-project.org). This implementation is highly optimized and written in the C programming language, allowing for very fast execution. Efficient implementations of Inspect [21] and the method of [15] are also available in the package.

Most similar to ESAC is the multiple changepoint detection procedure of Pilliat, Carpentier and Verzelen [15] for Gaussian changes in mean. The theoretical guarantees, for instance, are the same for ESAC and the method of [15]. Still, there are important distinctions between the two methods. As opposed to ESAC, the method proposed by [15] is based on their novel “bottom up” search. Their approach segments the data into disjoint segments chosen as narrow as possible from a predefined grid, where for each interval, a test statistic must have detected a changepoint. To ensure a disjoint segmentation, they merge overlapping segments of equal length whenever a changepoint is detected in both. From this segmentation, changepoint locations are estimated by taking midpoints of the segments. Consequently, the method of [15] only requires a test for a changepoint, and not a location estimator. In practice, this generality comes at a cost of changepoints being crudely estimated or not being detected at all, whenever the signal strength is low. This is illustrated in our simulation studies, which feature empirical comparisons between ESAC, the method of [15] and other proposed methods.

The paper is organized as follows. In Section 2 we give a formal description of the model assumed throughout the paper. In Section 3.1 we present a test statistic for a single changepoint that facilitates control over its family-wise error rate. In Section 3.2 we propose an estimator for the location of a single changepoint, also stating its finite sample estimation error rate with comparisons to other methods. In Section 3.3 we propose ESAC, which is our multiple changepoint estimation procedure. In Section 3.4 we present theoretical results regarding the statistical and computational properties of ESAC and compare these to other methods. In Section 4 we study the empirical performance of ESAC and other methods via simulations, including for misspecified models. In Section 5 we apply ESAC to sensor data from a Swedish hydro power plant. In Appendix A we prove our main theoretical results. In the remaining appendices we discuss

implementation of ESAC in practice, provide more simulation results, and prove auxiliary lemmas for our main results.

We use the following notation throughout the paper. For any vector $y \in \mathbb{R}^d$ we let y_j denote its j th component, $\|y\|_2$ denote its Euclidean norm and $\|y\|_0$ denote the number of nonzero entries in y . For any matrix $X \in \mathbb{R}^{p \times n}$ we let $X_{i,v}$ denote its (i, v) th element, $X_v \in \mathbb{R}^p$ denote its v th column, $X_{i,\cdot} \in \mathbb{R}^n$ denote its i th row, $\|X\|_F^2 = \sum_{i=1}^p \sum_{v=1}^n X_{i,v}^2$ denote the squared Frobenius norm of X , and $\|X\|_1 = \sum_{i=1}^p \sum_{v=1}^n |X_{i,v}|$ denote the entry-wise ℓ_1 norm of X . For any pair of matrices $X, Y \in \mathbb{R}^{p \times n}$, we let $\langle X, Y \rangle = \text{tr}(X^\top Y)$ denote their trace inner product. For any positive integer I we define $[I] = \{1, \dots, I\}$. For any pair of real numbers x, y , we define $x \vee y = \max\{x, y\}$ and $x \wedge y = \min\{x, y\}$. For any pair of random variables X, Y , we let $X \leq_{\text{st}} Y$ mean that Y stochastically dominates X , i.e. $\mathbb{P}(X \leq t) \geq \mathbb{P}(Y \leq t)$ for all $t \in \mathbb{R}$. For any $x \in \mathbb{R}$, we let $\lfloor x \rfloor$ denote the largest integer no larger than x , and $\lceil x \rceil$ denote the smallest integer no smaller than x .

We also find it useful to adopt the notation of $[1]$ to denote integer intervals. For any pair of integers s, e such that $s \leq e - 2$, we let (s, e) denote the open integer interval $\{s + 1, \dots, e - 1\}$ and let $(s, e]$ denote the left-open and right closed integer interval $\{s + 1, \dots, e\}$.

2. Problem description

To motivate our method and facilitate theoretical analysis, we consider the following model for the remainder of the paper. Note that we assess the performance of our model under deviations from this model in Section 4. Suppose we observe $n \geq 2$ independent multivariate Gaussian variables

$$X_v = \mu_v + W_v, \quad (1)$$

where $\mu_v \in \mathbb{R}^p$ and $W_v \sim N_p(0, \sigma^2 I)$ for $v \in [n]$. Assume that there are $J \geq 0$ changepoints $0 < \eta_1 < \dots < \eta_J < n$ such that

$$\mu_v \neq \mu_{v+1} \text{ if and only if } v = \eta_j \text{ for some } j \in [J].$$

Let $\theta_j = \mu_{\eta_{j+1}} - \mu_{\eta_j}$ denote the change in mean occurring at the j th changepoint, and let $\varphi_j = \|\theta_j\|_2$ be the ℓ_2 -norm of the mean-change occurring at changepoint j . Further, let $k_j = \|\theta_j\|_0$ denote the *sparsity* of the j th changepoint, i.e. the number of non-zero components of θ_j . Lastly, let $\Delta_j = \min(\eta_j - \eta_{j-1}, \eta_{j+1} - \eta_j)$ denote the minimum distance between the j th changepoint and a neighboring changepoint (where we for convenience take $\eta_0 = 0$ and $\eta_{J+1} = n$). Our goal is to estimate J , the number of changepoints, and their locations $\eta_1 < \dots < \eta_J$.

In the theoretical analysis to follow, we take σ^2 to be known. For compactness of notation, let $X, \mu \in \mathbb{R}^{p \times n}$ denote the matrices with X_v, μ_v as their v th columns, respectively, for $v \in [n]$.

3. Method and results

3.1. Single changepoint detection with family-wise error rate control

We begin by presenting a statistical test for a single change in mean in some arbitrary interval $(s, e) = \{s+1, \dots, e-1\}$, where $0 \leq s < e \leq n$, and $s \leq e-2$. To simplify the exposition, assume for now that $\sigma = 1$, as the data can be normalized to satisfy this assumption. We seek a test statistic that facilitates control over the family-wise error rate when testing for changepoints over multiple intervals. This level of control is needed later on, when we define a multiple changepoint algorithm in Section 3.3.

To construct a test statistic, we build on the work of Liu, Gao and Samworth [13]. They propose an efficient and minimax rate optimal test statistic for testing for a single changepoint within an interval. Unfortunately, this test does not allow for control of the family-wise error rate, and thus needs modification, presented next. For a direct comparison with the changepoint test in [13], see the end of this section.

For any candidate changepoint location v such that $0 \leq s < v < e \leq n$, we define the CUSUM transformation $T_{(s,e]}^v(y)$ of a vector $y \in \mathbb{R}^n$ as

$$T_{(s,e]}^v(y) = \left\{ \frac{e-v}{(e-s)(v-s)} \right\}^{1/2} \sum_{i=s+1}^v y_i - \left\{ \frac{v-s}{(e-s)(e-v)} \right\}^{1/2} \sum_{i=v+1}^e y_i. \quad (2)$$

To simplify notation, we use $C_{(s,e]}^v(i) = T_{(s,e]}^v(X_{i,\cdot})$ to denote the CUSUM of the i th component of the data within the integer interval $(s, e]$, evaluated at candidate changepoint position v .

Given a candidate sparsity level $t \in [p]$, and penalizing function $\gamma(t)$, both to be discussed later, define

$$S_{\gamma,(s,e]}^v(t) = \sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(t)} \right\} \mathbb{1} \left\{ |C_{(s,e]}^v(i)| \geq a(t) \right\} - \gamma(t), \quad (3)$$

where the threshold value $a(t)$ is given by

$$a^2(t) = 4 \log \left(\frac{ep \log n}{t^2} \right) \mathbb{1} \left\{ t \leq (p \log n)^{1/2} \right\}, \quad (4)$$

and $\nu_{a(t)}$ is a mean-centering term defined by taking $\nu_a = \mathbb{E}(Z^2 \mid |Z| \geq a)$ for $Z \sim \mathcal{N}(0, 1)$ and $a \geq 0$. In (4), we abuse notation slightly, writing $e = \exp(1)$ to mean Euler's number.

The CUSUM is a linear operation, and thus each of the CUSUMs $C_{(s,e]}^v(i)$ is the sum of the CUSUMs of the noise and the true means. As is common in sparse signal detection, the quantity in (3) thresholds these CUSUM in order to separate the signal from the noise. Here, the signal is made up of the CUSUM transformations of the true means, each of these being zero when no changepoint is present, and growing with the size of the change whenever a changepoint is

present. The thresholding value $a(t)$ is tailored specifically for a given sparsity level t , chosen just large enough to avoid that a t -sparse change drowns in the noise, and is decreasing in t . The term $\nu_{a(t)}$ in (3) serves as a mean-centering term for the few CUSUM values that spuriously exceed the threshold $a(t)$, and satisfies $a^2(t) \leq \nu_{a(t)} \leq a^2(t) + 2$. In an effort to borrow information across coordinates, the thresholded CUSUMs are then squared and summed. In spite of the thresholding and mean-centering taking place, this sum need not be small even when no changepoint is present, especially when the threshold $a(t)$ is small. The role of the penalty function $\gamma(t)$ is therefore to ensure that $S_{\gamma,(s,e]}^v(t) < 0$ with high probability whenever no changepoint is present.

We refer to $S_{\gamma,(s,e]}^v(t)$ as a *sparsity-specific penalized score*, which heuristically measures the degree of evidence of a changepoint at $v \in (s, e)$ for a fixed sparsity t . The true sparsity of the changepoint, however, is not known. To measure the overall degree of evidence of a changepoint at location v , we consider an exponentially increasing grid of candidate sparsity levels

$$\mathcal{T} = \{1, 2, 4, \dots, 2^{\lceil \log_2 \lfloor \sqrt{(p \log n)} \rfloor \rceil}\} \cup \{p\}. \quad (5)$$

This approach is also taken by [13], where the grid \mathcal{T} is slightly smaller. This choice of grid is justified as follows. Whenever a changepoint has true sparsity $k < (p \log n)^{1/2}$, there always exists some $t \in \mathcal{T}$ such that $t/2 \leq k \leq t$, which turns out to be sufficient for detecting the changepoint. Conversely, when the true sparsity k satisfies $k \geq (p \log n)^{1/2}$, it is sufficient to consider $t = p$.

For a candidate changepoint position $s < v < e$, define the *penalized score* as

$$S_{\gamma,(s,e]}^v = \max_{t \in \mathcal{T}} S_{\gamma,(s,e]}^v(t), \quad (6)$$

which heuristically measures the degree of evidence of a changepoint at location v , regardless of the sparsity.

As our test statistic for a changepoint in the interval (s, e) , we take

$$S_{\gamma,(s,e]} = \mathbb{1} \left\{ \max_{s < v < e} S_{\gamma,(s,e]}^v > 0 \right\}. \quad (7)$$

Note that we could also have maximized $S_{\gamma,(s,e]}^v$ over a geometric grid of vs , such as in [13], but this would not have led to any improvement in performance, save for a slight decrease in computational cost.

As for the penalty function $\gamma(t)$, for $t \in [p]$ define

$$r(t) = r(t, n, p) = \begin{cases} (p \log n)^{1/2} & \text{if } t \geq (p \log n)^{1/2}, \\ t \log \left(\frac{ep \log n}{t^2} \right) \vee \log n & \text{otherwise.} \end{cases} \quad (8)$$

With penalty function $\gamma(t) = \gamma_0 r(t)$ for some suitably large constant $\gamma_0 > 0$, we obtain the following control over the family-wise error rate.

Proposition 3.1. *Consider the model in Section 2. For all s, e and v such that $0 \leq s < v < e \leq n$, assume that the quantity $S_{\gamma, (s, e]}$ is computed with variance-scaled input matrix $\tilde{X} = (1/\sigma)X$ and penalty function $\gamma(t) = \gamma_0 r(t)$ for some $\gamma_0 > 0$. Let I denote the set of all intervals $(s, e) \subseteq (0, n)$ containing no changepoint, i.e. satisfying $\eta_j \notin (s, e) \forall j \in [J]$. For any $\varepsilon > 0$, there exists a universal choice of $\gamma_0 > 0$ (depending only on ε) such that*

$$\mathbb{P} \left(\max_{(s, e) \in I} S_{\gamma, (s, e]} > 0 \right) \leq \varepsilon.$$

Some remarks are in order. Figure 1 displays plots of $a^2(t)$, $\nu_{a(t)}$ and $r(t)$ as functions of t , for $n = p = 500$. As our first remark, we observe that $a(t)$ and $\nu_{a(t)}$ are decreasing in t , while $r(t)$ is increasing for all $t \leq (p \log(n)/e)^{1/2}$, but with a bulk when t is close to $(p \log n)^{1/2}$. Thus, the CUSUMs are thresholded more harshly when the candidate sparsity level t decreases, while the penalty function shrinks, at least when t is not close to $(p \log n)^{1/2}$. Note that several equivalent monotonic functions can be chosen in the place of $r(t)$, but we have chosen $r(t)$ due to its simple analytical form. The function $r(t)$ can be seen as the information theoretic detection boundary in terms of the Signal-to-Noise Ratio (SNR) for multiple changes in mean of sparsity t in p -dimensional Gaussian vectors with sample size n (see Section 3.4 or [15]). When $p = 1$, we recover the standard penalty used in the univariate changepoint literature for Gaussian changes in mean [22], as $r(1) = \log n$ in this case. As our second remark, the forms of the threshold $a(t)$ and penalty function $\gamma(t) \propto r(t)$ reflect the two sparsity regimes known in the statistical literature on multivariate mean change detection. In the sparse case where $t \leq (p \log n)^{1/2}$, the threshold $a(t)$ is non-zero and satisfies $a^2(t) \approx r(t)/t$, which is decreasing with t . Meanwhile, in the dense case where $t > (p \log n)^{1/2}$, the threshold satisfies $a(t) = 0$, in which case no thresholding takes place and all CUSUMs contribute to (3).

Lastly, we discuss the difference between the test in (7) and that of that of [13]. As the test in (7) is designed to be applied over several intervals, the thresholding in (7) is more stringent than that in [13]. To facilitate control over the family-wise error rate, the threshold $a(t)$, the mean-centering term $\nu_{a(t)}$, and the penalty function $\gamma(t) = \gamma_0 r(t)$ grow faster with n than their equivalent counterparts in [13]. To (approximately) recover the test statistic from [13], one must replace $\log n$ by $\log \log(8n)$ in (3), (4), (5), (6), (7) and (8), replace the set of candidate v 's with a dyadic grid, and use the penalty function $\gamma(t) = \gamma_0 r(t)$ with the modified function $r(t)$. To exactly recover the test in [13], the CUSUM must also be replaced by a CUSUM-like quantity defined in that paper.

3.2. Single changepoint estimation

We now consider the problem of estimating the location of a single changepoint within some interval (s, e) , assuming the changepoint has already been detected

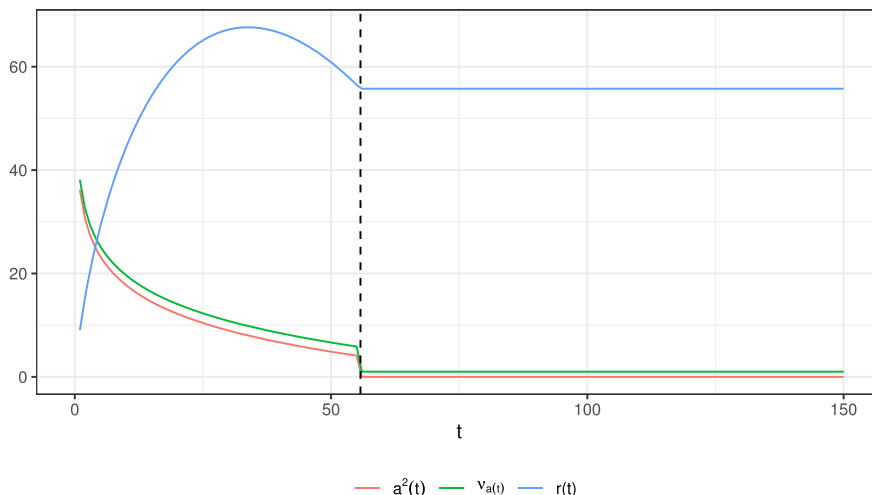


FIG 1. Plots of $a^2(t)$ (red), $\nu_{a(t)}$ (green), and $r(t)$ (blue) for $n = p = 500$. The boundary between the dense and sparse regimes is given by a vertical dashed line at $k = (p \log n)^{1/2}$.

or is known to be present. As before, we assume that $\sigma = 1$, as the data can be normalized to satisfy this assumption. Without loss of generality, recalling the model defined in (1), we may assume that $J = 1$ and $[s, e] = [0, n]$. To enhance readability, we will in the following suppress $[s, e]$ from the notation. The problem at hand is to estimate the location $\eta = \eta_1$ of a single changepoint, taking the sparsity $k = k_1$ as unknown.

In Section 3.1, we saw how the penalized score $S_\gamma^v = S_{\gamma, (0, n]}^v$ could be used to test for a changepoint at a location v in the interval $(s, e] = (0, n]$. In essence, this testing procedure is adaptive to the unknown sparsity by testing for a changepoint over a grid \mathcal{T} of candidate sparsity levels, using $S_\gamma^v(t) = S_{\gamma, (0, n]}^v(t)$ as a sparsity-specific test statistic. A novel methodological contribution of this paper is to recognize and prove that, with a suitable choice of penalty function, the penalized score also provides an accurate changepoint estimator, regardless of the true sparsity. As our changepoint estimator, we take the location v that maximizes the penalized score;

$$\hat{\eta}_\lambda = \arg \max_{0 < v < n} S_\lambda^v. \quad (9)$$

Note here that the penalty function γ is replaced by λ . In practice, we experience that the optimal choice of penalty function for changepoint estimation is slightly smaller than for changepoint testing. Thus, mainly for practical purposes, we allow the penalty function in (9) to be different than in (7). To ensure that $\hat{\eta}_\lambda$ is always well defined, we formally set $\hat{\eta}_\lambda$ be the smallest maximizer of the objective function, although we suppress this from the notation.

The finite sample properties of $\hat{\eta}_\lambda$ are given in Theorem 3.2, which holds whenever $\lambda(t) = \lambda_0 r(t)$ for sufficiently large $\lambda_0 > 0$. Before stating the Theo-

rem, we discuss the intuition behind the estimator. An inspection of the proof of Theorem 3.2 reveals that the penalized score approximately recovers the signal (the sum of squared CUSUMs) from the affected coordinates. This makes it an effective estimator, since the sum of squared CUSUMs is always largest when evaluated at the true changepoint location. The penalized score recovers the signal from the data as follows. By maximizing the sparsity-specific penalized score $S_\lambda^v(t)$ in t , a trade-off occurs between the thresholding value $a(t)$, which is decreasing in t , and the penalty function $\lambda(t)$, which is increasing for most values of t . When λ_0 is sufficiently large, the penalty function $\lambda(t)$ cancels all contributions to the sum in (3) from coordinates with constant means, leaving only the squared CUSUMs from the affected coordinates. These remaining contributions are (approximately) maximized when t is of the same order as the true sparsity k , due to the trade-off occurring between the thresholding and the penalty function. As a result, S_λ^v approximately recovers the sum of squared CUSUMs of the affected coordinates up to a bounded error term.

The following finite sample result shows that the estimator $\hat{\eta}_\lambda$ is adaptive to the unknown sparsity and gives a high-probability upper bound on the estimation error. Consider the model in Section 2, with only one changepoint η , with sparsity k and ℓ_2 norm φ . Let $\Delta = \min(\eta, n - \eta)$. Let $\hat{\eta}_\lambda$ be as in (9), when the sparsity-specific penalized score $S_\lambda^v(t) = S_{\lambda, (0, n]}^v$ from (3) is computed with variance-scaled input matrix $\tilde{X} = (1/\sigma)X$ and penalty function $\lambda(t) = \lambda_0 r(t)$, where $\lambda_0 > 0$. Define

$$h(t) = h(t, n, p) = \begin{cases} [p \{\log n \vee \log \log(ep)\}]^{1/2} & \text{if } t \geq (p \log n)^{1/2}, \\ t \log \left(\frac{ep \log n}{t^2} \right) \vee \log n & \text{otherwise.} \end{cases} \quad (10)$$

We then have the following.

Theorem 3.2. *There exist a universal choice of $\lambda_0 > 0$ and universal constants $C_0, C_1 > 0$ such that, if*

$$\frac{\varphi^2 \Delta}{\sigma^2} \geq C_0 h(k), \quad (11)$$

we have that

$$\mathbb{P} \left\{ |\hat{\eta}_\lambda - \eta| \leq C_1 \frac{\sigma^2}{\varphi^2} h(k) \right\} \geq 1 - \frac{1}{n}.$$

The SNR requirement in (11) implies that the absolute estimation error of $\hat{\eta}_\lambda$ is no larger than $C_1 h(k) \sigma^2 / \varphi^2 \leq (C_1 / C_0) \Delta < \Delta$ whenever the conditions of the Theorem holds. In particular, in the asymptotic regime where k, p, Δ and φ vary with n , the quantity $|\hat{\eta}_\lambda - \eta| / \Delta$ converges in probability to 0 as $n \rightarrow \infty$ whenever $(\varphi^2 \Delta) / \{\sigma^2 h(k)\}$ diverges with n , which is the notion of consistency considered in e.g. [22] and [3]. Similarly, if $\varphi^2 / \{\sigma^2 h(k)\}$ diverges with n , then $\hat{\eta}_\lambda$ converges in probability to η as $n \rightarrow \infty$. Note that Theorem 3.2 requires that the penalty function $\lambda(t)$ has a specific functional form. For practical choices of the penalty function $\lambda(t)$, we refer to Appendix B.

Some performance comparisons with related methods are in order. In the following we let $C > 0$ denote a generic constant. The Inspect method [21] obtains an error rate of $(\sigma^2/\varphi^2) \log \log n$, which is smaller than the rate in Theorem 3.2, albeit under the much stronger SNR condition that $\varphi^2 \Delta/\sigma^2 \geq C(n/\Delta)k \log(p \log n)$. The error rate of the method proposed by [10] is of even smaller order, σ^2/φ^2 , although under the even stronger SNR condition that $\varphi^2 \Delta/\sigma^2 \geq C\{k \log(p \vee n)\}^2$. For the Double CUSUM algorithm [3, Section 4] in the single changepoint case where $\sigma = 1$, the asymptotic SNR requirement for consistency implies that $\varphi^2 \Delta/(k \log^2 n) \rightarrow \infty$. By “consistency” we mean that $|\hat{\eta}_{\text{DC}} - \eta|/\Delta$ converges to 0 in probability, where $\hat{\eta}_{\text{DC}}$ is the Double CUSUM estimate of η .

In summary, both Inspect and the method of [10] have smaller error rates than the estimator in (9), and especially so for larger values of the sparsity k . These rates are even minimax rate optimal [see Proposition 3 in 21], but come at the cost of substantially stronger signal strength conditions than in Theorem 3.2. As such, the estimator in (9) is, at least from a theoretical perspective, better suited to deal with dense changepoints. To demonstrate this phenomenon, Figure 2 displays the SNR requirement of the changepoint estimator in (9), Inspect, Double CUSUM and the method of [10] as a function of k , when $n = p = 500$. As we seek to illustrate the dependence on k , and the SNRs are only defined up to constant factors anyways, each SNR requirement in Figure 2 is normalized to have value 1 for $k = 1$. In the left plot, the SNRs are plotted as function of k on linear scale, while on a log scale to the right. The boundary between the dense and sparse regimes is indicated by the vertical dashed line at $k = (p \log n)^{1/2}$.

Figure 2 displays a dramatic difference in the methods’ SNR requirements as a function of k , in which the SNR condition of ESAC grows much slower with k . This phenomenon is also apparent in our simulation studies, in which the performance of Inspect, the Double CUSUM and the method of [10] all deteriorate for larger values of k . We emphasize that Figure 2 is only informative about the dependence on the SNR conditions on the sparsity k , as the SNR conditions of the methods are only identified up to constant factors. Note also that the apparent bulk in the SNR requirement of our changepoint estimator is a result of keeping the mathematical expression simple, as remarked in Section 3.1.

Although the SNR requirement of the estimator in (9) grows slower with k than the competing estimators, the effect of the sparsity is still substantial. For fixed values of n and p , the function $h(k)$ is increasing for most values of k , implying that both the SNR condition and the estimation error increase in k . As an example, the error rate for estimating a changepoint with sparsity $k = 1$ is $(\sigma^2/\varphi^2) (\log n \vee \log p)$, while the same error rate becomes $(\sigma^2/\varphi^2) [p \{\log n \vee \log \log(ep)\}]^{1/2}$ for $k = p$.

Lastly, we remark that Theorem 3.2 hinges upon the assumption that the noise is isotropic Gaussian with no temporal dependence. In practice, this assumption is unrealistic. In Section 4, we investigate the performance of the estimator in (3.2), and other methods, via simulations. Here we consider various

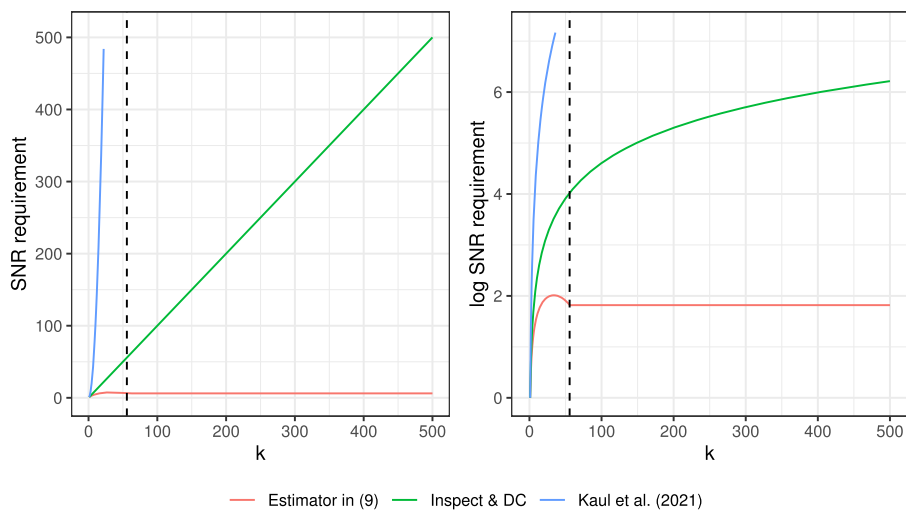


FIG 2. Normalized SNR conditions of the changepoint estimator in (9) (red), Inspect and Double CUSUM (green), and the method of [10] (blue), plotted as a function of the sparsity k , on a linear scale (left) and log scale (right). The boundary between the dense and sparse regimes is given by a vertical dashed line at $k = (p \log n)^{1/2}$.

model misspecifications, including temporal dependence and light- and heavy-tailed deviations from the Gaussian model.

3.3. Detection and estimation of multiple changepoints

We now consider the combined problem of detecting and estimating an unknown number of changepoints in the data X_1, \dots, X_n . That is, our goal is to estimate J , the number of changepoints, and $(\eta_1, \dots, \eta_J)^\top$, the changepoint locations.

Our proposed test statistic from Section 3.1 and changepoint estimator from Section 3.2 are designed for segments $(s, e]$ with at most a single changepoint. Hence, a search procedure is essential to allow for multiple changepoint search. Our choice of search procedure is a slight variant of Seeded Binary Segmentation [12]. In essence, the Seeded Binary Segmentation search procedure generates a deterministic set of intervals (which they call seeded intervals), in each of which a single changepoint is searched for. As a single changepoint may be detected within several distinct intervals, a choice must be made regarding which of these intervals is to be used for estimating its location. We have opted for the Narrowest-Over-Threshold [1] choice of changepoints, using the narrowest interval in which a changepoint is detected to estimate its location. Our modification of Seeded Binary Segmentation is minor; in our variant, the generation of intervals is controlled by two parameters, α and K . The parameter K controls the distance between the centers of two consecutive intervals of the same length, and the parameter α controls the growth rate of the interval lengths.

Our algorithm for generating seeded intervals is found in Appendix B (Algorithm 4).

Our proposed multiple changepoint estimation procedure, ESAC, is as follows. Let $\mathcal{M} = \{(s_m, e_m] ; m \in [M]\}$ denote an enumerated collection of candidate intervals. Let $\gamma(t), \lambda(t)$ denote the penalty functions used in the sparsity-specific penalized score (3), for changepoint detection and estimation, respectively. Given data matrix X , our proposed method is initiated by calling the recursive algorithm $\text{ESAC}(X, (0, n], \mathcal{M}, \emptyset, \gamma, \lambda)$, defined by Algorithm 1.

For the theoretical analysis of ESAC given in Section 3.4, we find it necessary to consider a slightly modified variant of the algorithm, defined by Algorithm 2 in Appendix B. In this variant, candidate changepoint locations are trimmed away in the recursive step, discarding them from future use to detect or estimate further changepoints. The trimming of changepoints is introduced as a necessary technical step for the proof of Theorem 3.3 to go through, specifically to ensure that previously discovered changepoints are not re-discovered. In practice, we find trimming to be unnecessary and even weakening of the performance. An even more modified variant of ESAC, given by Algorithm 3 defined in Appendix B, takes only the midpoint of an interval as the only candidate changepoint location when testing for a changepoint, in addition to interval trimming. In practice, the modification in Algorithm 3 results in a substantial decrease in run time at the cost of reduced detection power, although the theoretical results in the next subsection hold for this variant as well. In practical application, we thus recommend using Algorithm 1 over Algorithms 2 and 3. A simulation study comparing the variants of ESAC is found in Appendix D.

Algorithm 1 $\text{ESAC}(X, (s, e], \mathcal{M}, \mathcal{B}, \gamma, \lambda)$.

Input: A matrix of observations $X \in \mathbb{R}^{p \times n}$, an open integer interval (s, e) in which candidate changepoints are searched for, an enumerated collection $\mathcal{M} = \{(s_m, e_m] ; m \in [M]\}$ of M half open integer sub intervals of $(0, n]$, a set of already detected changepoints \mathcal{B} , and penalty functions $\gamma(t), \lambda(t)$.
Output: Set \mathcal{B} of already detected changepoints.

```

if  $e - s \leq 1$ 
  return  $\mathcal{B}$ 
set  $\mathcal{M}_{(s,e]} = \{m \in [M] : (s_m, e_m] \subset (s, e]\}$ 
set  $\mathcal{O}_{(s,e]} = \left\{ m \in \mathcal{M}_{(s,e]} : \max_{s_m < v < e_m} S_{\gamma, (s_m, e_m]}^v > 0 \right\}$ 
if  $\mathcal{O}_{(s,e]} = \emptyset$ 
  return  $\mathcal{B}$ 
set  $l^* = \min_{m \in \mathcal{O}_{(s,e]}} |e_m - s_m|$ 
set  $\mathcal{O}_{l^*} = \{m \in \mathcal{O}_{(s,e]} : |e_m - s_m| = l^*\}$ 
set  $m^* = \operatorname{argmax}_{m \in \mathcal{O}_{l^*}} \max_{s_m < v < e_m} S_{\lambda, (s_m, e_m]}^v$ 
set  $v^* = \operatorname{argmax}_{s_{m^*} < v < e_{m^*}} S_{\lambda, (s_{m^*}, e_{m^*}]}^v$ 
 $\mathcal{B} \leftarrow \mathcal{B} \cup \{v^*\}$ 
 $\mathcal{B} \leftarrow \text{ESAC}(X, (s, v^*], \mathcal{M}, \mathcal{B}, \gamma, \lambda)$ 
 $\mathcal{B} \leftarrow \text{ESAC}(X, (v^*, e], \mathcal{M}, \mathcal{B}, \gamma, \lambda)$ 
return  $\mathcal{B}$ 

```

3.4. Theoretical results for multiple changepoints

For the variants of ESAC defined by either Algorithm 2 or Algorithm 3 (both given in Appendix B), we have the following finite-sample statistical result. Let $X \in \mathbb{R}^{p \times n}$ follow the model in Section 2, and let $r(t)$ be defined as in (8). Let $\mathcal{M} = \{(s_m, e_m) ; m \in [M]\}$ denote the set of candidate intervals generated from Algorithm 4 with parameters $\alpha \leq 2$, $K \geq 2$, and let the penalty function $\gamma(t)$ be defined as $\gamma(t) = \gamma_0 r(t)$. Then we have the following.

Theorem 3.3. *There exists a universal choice of $\gamma_0 > 0$, such that for some universal constants $C_0, C_1 > 0$, depending only on γ_0 , and for any choice of $\lambda(t)$, if the SNR condition*

$$\frac{\varphi_j^2 \Delta_j}{\sigma^2} \geq C_0 r(k_j) \quad (12)$$

holds for all $j \in [J]$, we have that

$$\mathbb{P} \left\{ \hat{\mathcal{J}} = J \cap |\hat{\eta}_j - \eta_j| \leq C_1 \frac{\sigma^2}{\varphi_j^2} r(k_j) \quad \forall j \in [J] \right\} > 1 - \frac{1}{n}.$$

The explicit values of γ_0, C_0 and C_1 can be found in the proof of Theorem 3.3 in Appendix A, although we remark that these constants have not been optimized. Theorem 3.3 implies that the changepoint estimates outputted by ESAC have errors no larger than $C_1 \sigma^2 r(k_j) / \varphi_j^2 < \Delta_j$. In particular, in the asymptotic regime where J, k_j, p, Δ_j and φ_j vary with n , the quantity $\max_{j \in [J]} |\hat{\eta}_j - \eta_j| / \Delta_j$ converges in probability to 0 as $n \rightarrow \infty$ whenever $(\varphi_j^2 \Delta_j) / \{\sigma^2 r(k_j)\}$ diverges with n for all $j \in [J]$. Similarly, if $\varphi_j^2 / \{\sigma^2 r(k_j)\}$ diverges with n for all $j \in [J]$, then $\max_{j \in [J]} |\hat{\eta}_j - \eta_j|$ converges in probability to 0 as $n \rightarrow \infty$.

We now clarify the role of the estimation step in ESAC. Notice that Theorem 3.3 holds for any choice of λ , and that the error rate in Theorem 3.3 is slightly smaller than in Theorem 3.2 for dense changepoints. This is because the estimation error rate implied by Theorem 3.3 is a consequence of the testing step in Algorithm 2 (and in Algorithm 3), and thus the estimation steps in ESAC does not contribute to the theoretical bound on the estimation error in Theorem 3.3. Indeed, due to the Narrowest-over-Threshold choice of changepoints, it is sufficient to bound the largest width of a candidate interval, an observation due to [15], which in our case is of the order of $\sigma^2 r(k_j) / \varphi_j^2$ for the j th changepoint. In particular, Theorem 3.2 is not a corollary of Theorem 3.3. In fact, the estimation error in Theorem 3.3 is surprisingly lower than in Theorem 3.2, since $h(k) \geq r(k)$ whenever $k \geq (p \log n)^{1/2}$. We consider the discrepancy between the two rates to be an artifact of our proof techniques. In practice, the estimation step occurring after the testing step substantially improves the estimation error of the changepoint locations, in comparison with taking e.g. the midpoint of the interval in which a changepoint is detected. For more details and an empirical investigation, we refer to Appendix D.

To minimize the estimation error of ESAC, we recommend choosing the penalty functions $\gamma(t)$ via Monte Carlo simulation or setting $\lambda(t), \gamma(t)$ proportional to a slight variant of $r(t)$. In particular, when using $\lambda(t), \gamma(t) \propto r(t)$, our simulations suggest that the leading constants can be chosen independently of n and p , when model assumptions are satisfied, at least for the values of n and p we have considered. The assumptions required in Theorem 3.3 are undeniably strong, requiring temporally and spatially independent isotropic noise, which can be difficult to meet in practice. In such cases, we recommend choosing $\lambda(t)$ and $\gamma(t)$ via Monte Carlo simulation, where the errors are sampled from a (possibly temporally and spatially dependent) heavy-tailed distribution, such as in the real data example in Section 5. For further details and recommendations regarding the choice of penalty functions, we refer to Appendix B.

Some performance comparisons to related methods are in order. In the following, we let $C > 0$ denote a generic constant. To begin, Theorem 3.3 gives a very similar theoretical guarantee as the method of Pilliat, Carpentier and Verzelen [15, Corollary 3]. In fact, when the probability of the desired event in the Corollary is the same as in Theorem 3.3 (i.e. setting $\delta = 1/n$ in the Corollary), the method of Pilliat obtains the same error rate under an up to constants equal SNR requirement. The Inspect method of [21] obtains an error rate of $(\sigma^2/\varphi^2)(n/\Delta)^4 \log(np)$, where $\varphi = \min_{j \in [J]} \varphi_j$, $\Delta = \min_{j \in [J]} \Delta_j$ and $k = \max_{j \in [J]} k_j$. The error rate of ESAC is therefore smaller than that of Inspect whenever n/Δ is large (short distance between changepoints) or k is sufficiently large. The SNR condition needed for the error rate of Inspect to hold is that $\varphi^2 \Delta / \sigma^2 \geq C \log(np) \{(n/\Delta)^3 \vee k\} (n/\Delta)$, which is stronger than that of ESAC, and especially so when k is large or the changepoints are close to each other. The method of Kaul and Michailidis [9] obtains an error rate of $(\sigma^2/\varphi^2) \log^2(n)$, which is mostly smaller than that of ESAC. The method requires the SNR condition that $\varphi^2 \Delta / \sigma^2 \geq CJ^2 k^2 \log^3(n \vee p)$, which is substantially stronger than the SNR condition of ESAC, especially if there are many changepoints or k is large. For the Double CUSUM algorithm [3], in the case where $\sigma = 1$, the error rate is at least of the order $\log^2(n) k_j / \varphi_j^2$ for the j th changepoint, which is larger than that of ESAC. The SNR condition for the Double CUSUM algorithm is that $\varphi_j^2 \Delta_j / (n^{3-5\beta/2} k_j \log^2 n) \rightarrow 0$ and $n^\beta = \mathcal{O}(\Delta_j)$ for all $j \in [J]$, for some $\beta \in (6/7, 1]$, as well as p being of the same order as n^ω for some fixed $\omega > 0$, which is uniformly stronger than that of ESAC.

In summary, the error rate of ESAC is smaller than that of the Double CUSUM algorithm for all values of k , mostly larger than the method of Kaul and Michailidis [9], and only smaller than that of Inspect for very small values of k . These smaller error rates displayed by Inspect and [9] come at a cost of substantially larger signal strength conditions, which grow much faster with k . To illustrate this phenomenon, Figure 3 displays the SNR requirements of ESAC (red), Inspect (green), the Double CUSUM algorithm (blue) and the method of [9] (black) on a log scale as a function of the sparsity k , plotted for different values of n and p . In the left plot, the log SNR requirements are plotted for $n = 10^2, 10^3, 10^4$ with $p = 500$ fixed. To the right, they are plotted for $p = 500, 1000, 2000$, keeping $n = 500$ fixed. Each SNR requirement is normalized

to have value 1 for sparsity $k = 1$ at $n = 100$ (left) and at $p = 500$ (right), as the SNR conditions are only identified up to constant factors anyway. For Inspect, we have set $\Delta = n/2$, and for the method of [9] we have set $J = 1$, which is to these methods' advantage. Note that in the left plot, the log SNR conditions of Inspect and Double CUSUM overlap for $n = 10^2$. In the right plot, the log SNR conditions for Inspect and Double CUSUM are very close over all considered values of p , making them difficult to tell apart. As before, we emphasize that the curves in Figure 3 only illustrate the dependence of the SNR conditions on n, p and k , and the curves themselves should not be compared to each other directly.

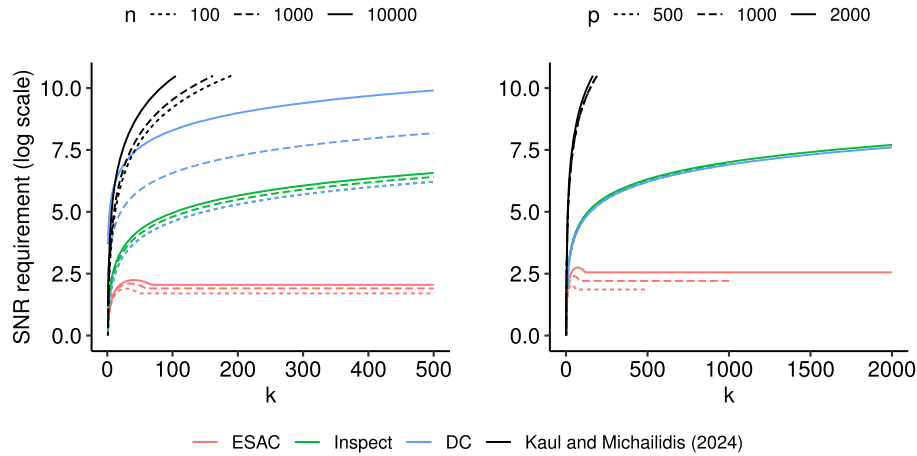


FIG 3. Normalized SNR conditions of ESAC (red) and Inspect (green) and the Double CUSUM algorithm (blue) on a log scale, plotted as a function of the sparsity k , and varying values of n (left) and p (right).

We now turn to optimality considerations. Observe first that the SNR condition for ESAC in (12) is up to constants minimal for identifying J , the number of changepoints. Indeed, for any n, p and $k \leq p$, an implication of Theorem 2 in Pilliat, Carpentier and Verzelen [15] is that

$$\sup_{P \in Q(n,p,k)} P(|\hat{\eta}| \neq J) \geq 1/4$$

for all estimators $\hat{\eta} = \hat{\eta}(X_1, \dots, X_n)$ of the changepoint vector $\eta = (\eta_1, \dots, \eta_J)$, where $Q(n, p, k)$ is the class of all probability distributions of X_1, \dots, X_n corresponding to the model given in Section 2 for which $k_j \leq k$ and $\varphi_j^2 \Delta_j / \sigma^2 \geq cr(k_j)$ for all $j \in [J]$ and for some sufficiently small $c > 0$. In comparison, ESAC is guaranteed to correctly estimate J with high probability whenever $\varphi_j^2 \Delta_j / \sigma^2 \geq Cr(k_j)$ for all $j \in [J]$ and some sufficiently large $C > 0$. As for the changepoint location error rate, the minimax rate has been shown by Wang and Samworth [21] to be at least $\sigma^2 / (16\varphi^2)$ whenever $\Delta^{-1} \leq \varphi^2 / \sigma^2 \leq 1$. Hence, at least in this region of the parameter space, the estimator $\hat{\eta}$ from Section 3.2 and the

full ESAC algorithm have minimax optimal error rates up to factors of $h(k)$ and $r(k)$, respectively, where k is the sparsity of the changepoint in question. Note that, while $r(k)$ and $h(k)$ are constant multiples of $\log n$ whenever $k = 1$, they grow substantially with k . Hence the error rate of ESAC is only close to minimax rate optimal for small values of the sparsity k .

Finally, we consider the computational cost of ESAC as a function of the size of the data. The following Proposition shows that ESAC has a log-linear computational cost.

Proposition 3.4. *Consider any input matrix $X \in \mathbb{R}^{p \times n}$, penalty functions $\gamma(t), \lambda(t)$ and seeded intervals generated by Algorithm 4 with fixed input parameters $\alpha > 1$ and $K \in \mathbb{N}$. Then the computational complexity of ESAC (either Algorithm 1, 2 or 3) measured in floating point operations is of order $\mathcal{O}\{np \log(p \log n)\}$ in the best case and $\mathcal{O}\{np \log n \log(p \log n)\}$ in the worst case.*

In Proposition 3.4, the best case computational cost of ESAC occurs when there are $n - 1$ detected changepoints, and the worst case computational cost occurs when there are no changepoints detected. In comparison, the computational complexity of the method of Pilliat, Carpentier and Verzelen [15] is $\mathcal{O}\{np \log(np)\}$, which is of slightly smaller order than the worst-case complexity of ESAC. In our simulation study, however, we experience that the computational costs of ESAC and the method in [15] have very similar dependence on n and p , with ESAC being faster by a seemingly constant factor. For the other multiple changepoint methods like the Double CUSUM, Sparsified Binary Segmentation, SUBSET and Inspect, no specific forms of computational cost are provided in the respective articles. For an empirical comparison of run times, we refer to the next section.

4. Simulations

We now compare the empirical performance of ESAC and our single changepoint estimator in (9) with the following state-of-the-art methods for high-dimensional changepoint detection and estimation: a variant of the Inspect method by Wang and Samworth [21], the method of Pilliat, Carpentier and Verzelen [15] hereby called Pilliat, Sparsified Binary Segmentation of Cho and Fryzlewicz [4], the Double CUSUM algorithm of Cho [3], the SUBSET method by Tickle, Eckley and Fearnhead [19], and the methods of Kaul et al. [10] and Kaul and Michailidis [9]. We introduce a slightly modified variant of Inspect, based on Narrowest-over-Threshold search, mainly to reduce computational cost. The details of our modified Inspect algorithm can be found in Appendix C. To run the Sparsified Binary Segmentation and Double CUSUM algorithms, we use the R package **hdbinseg** [5]. To run SUBSET, we use the code from the Github repository of Tickle [18]. To run the methods of [10] and [9], we use the code provided to us by the first author of these two publications. We have implemented the remaining methods ESAC, Pilliat and Inspect in the C programming language, which are found in the R package **HDCD** [14], available on CRAN. We remark that our

implementations of Inspect and the Pilliat method are orders of magnitude faster than their original implementations. Replication code for our simulations studies can be found in the `/inst` folder in the package. Whenever run times are reported, they have been run using R (4.2.1) on a MacOS (12.3) computer with an (ARM) Apple M1 Pro CPU.

For each method in the simulation study, a choice of penalty parameters must be made, which is discussed in each subsection. In all simulations, changes in mean are taken to have magnitudes spread evenly across all affected coordinates. In Appendix E we run the same simulations with uneven and random magnitudes, giving similar results. In all simulations we assume $\sigma = 1$ is unknown. We estimate σ separately for each of the p coordinates of the observed time series, and use it to normalize the data before applying each changepoint detection method. As is commonly done in the changepoint literature, we estimate the noise level by the median absolute deviation of first-order differences with scaling factor 1.05 for the Gaussian distribution.

4.1. Single changepoint estimation

We first consider the algorithms' performance when estimating the location of a single changepoint, assuming that it has already been detected. Our simulations are run with parameters $n \in \{200, 500\}$, $p \in \{100, 1000, 5000\}$, $k \in \{1, \lceil p^{1/3} \rceil, \lceil \sqrt{p \log n} \rceil, p\}$. For each configuration of these parameters, we simulate 1000 data sets and apply the methods considered in the study. For each combination of n, p, k , the simulated data sets have a changepoint at $\eta = \lceil n/5 \rceil$ with change-vector $\theta \propto (I_1, \dots, I_k, 0, \dots, 0)^\top$, where I_1, \dots, I_k are drawn independently and uniformly from $\{-1, 1\}$. For each sample we scale θ such that $\Delta \varphi^2 = (n/5) \|\theta\|_2^2 = (5/2)^2 r(k)$, where k is the sparsity of the change and $\Delta = \lceil n/5 \rceil$. For a simulation study in which the change-vector θ is drawn randomly, see Appendix E.2.

To keep the simulation study simple, we use the authors' recommended non-empirical choices of penalty parameters. We take ESAC to be the estimator given in (9), with penalty function $\tilde{\lambda}(t)$ as defined in Appendix B. As for Inspect, we use Algorithm 2 in Wang and Samworth [21], with penalty parameter $\lambda = \{\log(p \log n) / 2\}^{1/2}$. For the Double CUSUM algorithm we set $\varphi = -1$, corresponding to the version presented in Section 4.1 of Cho [3]. For the method of [10], we use the default parameters provided in code from that paper's simulation study. For Sparsified Binary Segmentation, a default choice of the threshold π_T is not available, so we take π_T to be the maximum value of the CUSUMs $|T_{[0,n]}^v(Z_{i,\cdot} / \hat{\sigma}_i)|$ over all values of $0 < v < n$ and $i \in [p]$, where $Z_{v,i} \sim N(0, 1)$ independently for $i \in [p]$, $v \in [n]$, and $\hat{\sigma}_i$ is the median absolute deviation of the noise level in the i th series, based on 1000 Monte Carlo samples. Whenever the Sparsified Binary Segmentation estimator is not defined, we set its output to be 1. For both the Double CUSUM and Sparsified Binary Segmentation algorithm, we specify `height = 1` when calling the respective functions to turn the methods into single changepoint estimators. The Pilliat method is not in-

cluded in this simulation as there is no straightforward way to modify it into a single changepoint estimator. The method of [9] is not included either as it is a multiple changepoint extension of the single changepoint method found in [10].

TABLE 1
Single changepoint estimation MSE.

Parameters					Mean Squared Error					
n	p	k	η	φ	ESAC	Inspect	SBS	SUBSET	DC	Kaul et al.
200	100	1	40	1.40	10.4	25.3	83.0	54.2	9.8	59.4
200	100	5	40	2.00	5.8	4.1	389.0	3.0	13.8	9.1
200	100	24	40	1.90	96.5	139.6	1495.8	251.1	931.1	777.0
200	100	100	40	1.90	95.1	425.9	1520.6	250.8	2719.4	13621.7
200	1000	1	40	1.52	6.9	105.8	29.5	33.1	6.5	8.9
200	1000	10	40	2.93	5.1	0.8	130.6	1.2	9.2	0.5
200	1000	73	40	3.37	4.6	64.5	1478.2	8.2	163.7	1.6
200	1000	1000	40	3.37	3.5	796.2	1534.7	7.1	207.5	18379.5
200	5000	1	40	1.60	45.3	413.3	29.1	81.4	142.1	7.9
200	5000	18	40	4.00	9.4	0.6	65.7	3.3	84.2	0.2
200	5000	163	40	5.04	3.6	60.3	1466.2	3.6	7.1	4.3
200	5000	5000	40	5.04	4.4	1453.9	1563.1	4.4	5.3	19780.0
500	100	1	100	0.92	55.4	97.9	120.8	216.4	54.7	63.7
500	100	5	100	1.31	22.9	15.7	1060.1	12.6	100.6	41.5
500	100	25	100	1.25	112.7	323.1	9560.3	2150.5	6420.7	4084.3
500	100	100	100	1.25	284.4	1845.9	9768.9	1959.7	17072.2	121078.1
500	1000	1	100	1.00	30.5	217.7	79.3	190.5	31.0	36.9
500	1000	10	100	1.90	12.3	5.1	233.0	3.7	78.1	3.8
500	1000	79	100	2.22	22.1	122.5	9547.8	66.4	1610.9	3.3
500	1000	1000	100	2.22	15.4	3895.9	9790.4	82.9	2091.1	141517.6
500	5000	1	100	1.05	22.2	1322.6	51.6	95.9	258.0	36.4
500	5000	18	100	2.58	7.9	1.8	103.1	3.2	627.6	1.4
500	5000	177	100	3.32	11.2	175.0	9438.2	11.2	37.5	1.3
500	5000	5000	100	3.32	20.2	8212.5	9799.4	33.7	28.0	146917.9
Average MSE					37.8	821.9	2889.1	230.3	1362.9	19434.9

For each method and each configuration of parameters, Table 1 displays the average Mean Squared Error (MSE), while Table 2 displays the average run time in milliseconds. In the tables, the Double CUSUM and Sparsified Binary Segmentation methods are abbreviated as DC and SBS, respectively. For each configuration of parameters, the minimum value of both the MSE and the run time are indicated in boldface. In terms of statistical accuracy, Table 1 demonstrates that ESAC and SUBSET are the only methods with competitive accuracy across the sparsity regimes, although ESAC has a slight edge over SUBSET. ESAC has the lowest MSE in 11 out of the 24 different combinations of parameters (including both dense and sparse regimes), while SUBSET has the lowest MSE in 5 out of 24, and the method of [10] has the lowest MSE in 7. When averaging the MSE over all the rows, ESAC is the clear winner, with SUBSET in second place. In comparison, the estimation accuracy of the method of [10] is excellent for $k \in \{1, \lceil p^{1/3} \rceil\}$, but deteriorates for higher sparsity levels, as does Inspect. The Double CUSUM algorithm displays excellent estimation accuracy when $k = 1$, but often not so for dense changepoints (although this seems to

vary slightly with n and p). Sparsified Binary Segmentation has a decent estimation accuracy for sparse changepoints (especially when $k = 1$), but the accuracy deteriorates for dense changepoints.

TABLE 2
Single changepoint estimation run time.

Parameters					Time in milliseconds					
n	p	k	η	φ	ESAC	Inspect	SBS	SUBSET	DC	Kaul et al.
200	100	1	40	1.40	0.4	2.8	13.3	6.0	17.0	57.6
200	100	5	40	2.00	0.3	2.7	11.9	2.4	14.9	49.5
200	100	24	40	1.90	0.3	2.7	11.3	1.9	18.7	47.4
200	100	100	40	1.90	0.3	2.7	11.3	2.1	18.2	46.2
200	1000	1	40	1.52	2.2	57.9	107.8	17.1	149.6	318.9
200	1000	10	40	2.93	2.2	57.5	106.7	16.0	149.9	309.2
200	1000	73	40	3.37	2.2	58.0	105.4	15.8	149.8	305.5
200	1000	1000	40	3.37	2.2	57.8	105.5	16.3	149.2	300.1
200	5000	1	40	1.60	15.4	302.2	548.6	102.3	782.6	1543.4
200	5000	18	40	4.00	15.4	301.6	542.9	96.4	789.9	1529.5
200	5000	163	40	5.04	15.5	298.6	534.8	95.9	778.2	1518.8
200	5000	5000	40	5.04	15.3	301.3	530.5	95.3	778.6	1515.6
500	100	1	100	0.92	0.8	7.4	20.4	4.3	30.6	105.9
500	100	5	100	1.31	0.7	7.4	18.9	4.0	28.0	102.5
500	100	25	100	1.25	0.7	7.6	19.0	4.3	28.5	101.4
500	100	100	100	1.25	0.7	7.4	19.2	4.1	28.0	100.7
500	1000	1	100	1.00	6.2	357.0	174.3	38.6	292.5	713.4
500	1000	10	100	1.90	6.7	356.2	173.2	37.7	293.0	706.8
500	1000	79	100	2.22	7.1	356.1	170.4	38.2	292.0	703.6
500	1000	1000	100	2.22	7.2	356.6	172.2	38.2	291.8	705.1
500	5000	1	100	1.05	38.7	1839.1	895.2	206.8	1726.4	3427.7
500	5000	18	100	2.58	39.0	1829.9	868.0	206.5	1611.3	3420.7
500	5000	177	100	3.32	38.7	1834.6	869.8	202.5	1626.1	3393.5
500	5000	5000	100	3.32	40.6	1848.6	875.3	222.8	1631.2	3510.2
Average run time					10.8	427.2	287.7	61.5	486.5	1022.2

In terms of run time, ESAC is the clear winner, with overall execution time of around one sixth of SUBSET, the runner up, and down to around 1% of the execution time of the method of [10]. Note that SUBSET and the method of [10] are the only methods not implemented in C or C++, giving them a disadvantage when comparing run times. We also remark that the run time of scaling the data by the median absolute deviations is not included in the run times of ESAC, Inspect, SUBSET and the method of [10], as it would otherwise dominate the run time. The run time of the scaling is included in the running times of the Double CUSUM and Sparsified Binary Segmentation algorithms, as the implementations of these algorithms do not offer an option to disable it.

4.2. Multiple changepoint estimation

We now consider the situation of an unknown number of changepoints. Our simulations are run with parameters $n \in \{100, 200\}$, $p \in \{100, 1000\}$ and

$J \in \{0, 2, 5\}$. For each simulated data set we take the changepoint locations η_1, \dots, η_J to be ordered and uniformly drawn samples from $\{1, \dots, n-1\}$ without replacement. For each combination of n, p and J , we consider three different sparsity regimes; *dense*, *sparse* and *mixed*. In the dense and sparse regimes, we sample k_1, \dots, k_J independently and uniformly from $\{\lceil \sqrt{(p \log n)} \rceil, \dots, p\}$ and $\{1, \dots, \lfloor \sqrt{(p \log n)} \rfloor\}$, respectively. In the mixed regime we sample each k_j independently from a mixture between the dense and sparse regimes, each with equal probability. For each combination of n, p, J and sparsity regime, each changepoint has change-vector $\theta_j \propto (I_{j,1}, \dots, I_{j,k_j}, 0, \dots, 0)^\top$, where $I_{j,1}, \dots, I_{j,k_j}$ are drawn independently and uniformly from $\{-1, 1\}$, scaled such that $\Delta_j \varphi_j^2 / \sigma_j^2 = 4^2 r(k_j)$. Notice that we have increased the signal strength slightly in comparison with the single changepoint case, as multiple changepoint estimation is more challenging than estimating the position of a single changepoint whose existence is known. For each combination of n, p, J and sparsity regime we simulate 1000 data sets.

For both ESAC and the modified Inspect algorithm, we generate seeded intervals using Algorithm 4 with parameters $\alpha = 3/2$ and $K = 4$. For the Pilliat method we generate intervals using Algorithm 4 with parameters $\alpha = 3/2$ and $K = 2$, giving very similar intervals as the a -adic grid \mathcal{G}_a defined in [15] for $a = 2/3$. To run the method of [9], we use the output from Inspect as preliminary estimates of the changepoints, as this is the preliminary estimate used in the simulation study in [9]. Whenever the code from this article runs into an error, we set its output to be the preliminary estimates from Inspect. Note that the computation time of Inspect is included in the reported run time of the method of [9]. Due to high computational cost, we run SUBSET with only 100 randomly drawn intervals in its Wild Binary Segmentation step. To ensure comparability with the remaining methods, we have modified the Pilliat method so that its tests for a changepoint in an integer interval $(s, e]$ are performed by testing for a changepoint at each candidate position $s < v < e$, instead of only the mid-point. In our experience, testing only at the mid-point of an interval results in substantially lower detection power.

We choose detection thresholds for Sparsified Binary Segmentation and Double CUSUM using bootstrapping with $B = 100$ bootstrap samples. For the remaining methods, we choose detection thresholds using Monte Carlo simulations based on $N = 1000$ samples, with a target false positive probability of $\varepsilon = 1/100$. For the ESAC algorithm, we use the penalty functions $\tilde{\gamma}(t)$ and $\tilde{\lambda}(t)$ given in Appendix B, which is obtained via Monte Carlo simulation and a Bonferroni correction. For the Pilliat algorithm we choose detection thresholds for the Partial Sum statistic and the dense statistic by Monte Carlo simulating the leading constant in the theoretical thresholds given in [15], and apply a Bonferroni correction. For the modified version of Inspect we set $\lambda = \{\log(p \log n)/2\}^{1/2}$ and choose the detection threshold ξ to be the $N\varepsilon$ th largest sparse projection over all seeded intervals and over $N = 1000$ data sets with no changepoints. For SUBSET we use the function for choosing thresholds provided by the author, which is based on Monte Carlo simulation. For Sparsified Binary Segmentation and

Double CUSUM we use the default parameters when running the algorithms (except for setting $\varphi = -1$ for the Double CUSUM algorithm corresponding to the version presented in Section 4.1 of [3]) and use the default bootstrap procedures to select detection thresholds. Due to prohibitively high computational cost, the Double CUSUM algorithm is only run for $p = 100$.

For each method considered and each configuration of parameters and changepoint regimes, Table 3 displays the average Hausdorff distance between the true and estimated changepoints, as well as the average absolute estimation error of J in parenthesis. Note that the Hausdorff distance is only well defined when the true number of changepoints is nonzero. Note also that the Double CUSUM and Sparsified Binary Segmentation methods are abbreviated as DC and SBS, respectively. For each configuration of parameters and changepoint regimes, the minimum value of each of the two performance measures is indicated in boldface. In terms of average Hausdorff distance, Table 3 demonstrates that ESAC is the top performer in the simulation study, obtaining a smallest average Hausdorff distance in 23 out of the 26 parameter configurations with a changepoint. SUBSET obtains a second place, with performance arguably comparable to ESAC. For estimating J , ESAC is also the clear winner of the study, having the smallest estimation error in all configurations of the parameters with a changepoint present. Inspect obtains a second place, and consequently also the method of [9]. Note, however, that the latter method takes the estimated J from Inspect as an input, which is the reason their method has the same estimation error of J as Inspect. An extended version of Table 3, also including $p = 5000$, is given in Appendix E.1.

As for computational costs, Figure 4 displays the natural logarithm of the run times (in milliseconds) of the methods as functions of n and p , based on averages over $N = 24$ runs in the mixed sparsity regime with $J = 2$ changepoints. In the left plot, we fix $p = 100$ and let $n \in [100, 1000]$ vary, and in the right plot we fix $n = 100$ and let $p \in [100, 1000]$ vary. In terms of run time, ESAC outperforms the competing methods by a significant margin for all considered values of n and p . The run time of ESAC is smaller than that of the competitors by a factor seemingly constant in n and p . When not applying a log transform to the run times (which is omitted for brevity), all methods can be seen to have an approximately linear computational cost in both n and p .

4.3. Misspecified model

ESAC is designed for data with isotropic Gaussian noise, which can be an unrealistic assumption in practice. We now investigate the empirical performance of the changepoint estimator in (9) and the competing methods in the single changepoint setting under other data generating mechanisms than the model described in Section 2. We set $n = p = 200$. With the changepoint location fixed at $\eta = \lceil n/5 \rceil = 40$, we consider two sparsity regimes, *sparse* and *dense*. We sample k independently and uniformly from $\{1, \dots, \lfloor \sqrt{(p \log n)} \rfloor\}$ in the sparse regime, and from $\{\lceil \sqrt{(p \log n)} \rceil, \dots, p\}$ in the dense regime. In both regimes, we

TABLE 3
Multiple changepoints, Hausdorff distance and estimation error of J .

Parameters				Hausdorff distance ($ \hat{J} - J $)						
n	p	Sparsity	J	ESAC	Pilliat	Inspect	SBS	SUBSET	DC	Kaul et al
100	100	-	0	-(0.01)	-(0.00)	-(0.01)	-(0.02)	-(0.04)	-(0.01)	-(0.01)
100	100	Dense	2	0.76 (0.01)	9.40 (0.40)	2.19 (0.04)	41.77 (1.19)	1.08 (0.08)	45.30 (1.30)	2.19 (0.04)
100	100	Sparse	2	0.63 (0.01)	4.90 (0.19)	1.23 (0.03)	41.07 (1.09)	0.61 (0.04)	15.77 (0.46)	1.16 (0.03)
100	100	Mixed	2	0.53 (0.00)	7.75 (0.31)	2.11 (0.05)	41.01 (1.14)	1.23 (0.06)	32.48 (0.94)	2.26 (0.05)
100	100	Dense	5	0.48 (0.01)	10.53 (1.02)	2.87 (0.14)	45.06 (3.77)	1.35 (0.19)	34.02 (3.24)	2.72 (0.14)
100	100	Sparse	5	0.37 (0.01)	5.26 (0.43)	1.64 (0.11)	46.44 (3.76)	1.06 (0.21)	19.37 (2.13)	1.57 (0.11)
100	100	Mixed	5	0.43 (0.01)	7.75 (0.72)	2.61 (0.14)	45.64 (3.76)	1.19 (0.22)	25.39 (2.67)	2.51 (0.14)
100	1000	-	0	-(0.00)	-(0.00)	-(0.01)	-(0.27)	-(0.05)	-(0.01)	-(0.01)
100	1000	Dense	2	0.36 (0.00)	5.20 (0.24)	1.58 (0.03)	35.46 (0.98)	0.42 (0.03)	-(0.01)	9.18 (0.03)
100	1000	Sparse	2	0.30 (0.00)	3.88 (0.16)	3.02 (0.10)	39.76 (1.05)	0.38 (0.04)	-(0.01)	3.07 (0.10)
100	1000	Mixed	2	0.48 (0.01)	4.29 (0.18)	2.38 (0.06)	40.65 (1.05)	0.58 (0.10)	-(0.01)	6.20 (0.06)
100	1000	Dense	5	0.25 (0.00)	6.06 (0.56)	2.08 (0.10)	42.13 (3.62)	0.64 (0.17)	-(0.01)	2.19 (0.10)
100	1000	Sparse	5	0.23 (0.00)	3.85 (0.32)	3.09 (0.24)	44.72 (3.71)	0.65 (0.16)	-(0.01)	3.07 (0.24)
100	1000	Mixed	5	0.22 (0.00)	4.41 (0.38)	2.86 (0.19)	46.31 (3.77)	0.78 (0.17)	-(0.01)	2.98 (0.19)
200	100	-	0	-(0.01)	-(0.01)	-(0.01)	-(0.04)	-(0.06)	-(0.01)	-(0.01)
200	100	Dense	2	1.25 (0.01)	16.44 (0.38)	3.10 (0.03)	58.53 (0.89)	2.10 (0.07)	65.00 (0.98)	4.89 (0.03)
200	100	Sparse	2	0.97 (0.00)	7.07 (0.17)	1.87 (0.02)	47.77 (0.67)	1.26 (0.04)	12.07 (0.19)	1.72 (0.02)
200	100	Mixed	2	1.09 (0.00)	10.94 (0.25)	2.76 (0.02)	51.05 (0.75)	1.45 (0.03)	45.49 (0.69)	5.03 (0.02)
200	100	Dense	5	1.14 (0.01)	16.99 (1.01)	4.44 (0.09)	57.71 (3.02)	2.26 (0.19)	51.17 (2.64)	4.07 (0.09)
200	100	Sparse	5	0.89 (0.01)	7.57 (0.35)	2.19 (0.06)	56.82 (2.77)	2.02 (0.22)	21.56 (1.28)	2.02 (0.06)
200	100	Mixed	5	0.74 (0.00)	13.21 (0.69)	2.74 (0.07)	60.62 (2.93)	2.16 (0.20)	39.22 (2.09)	2.42 (0.07)
200	1000	-	0	-(0.00)	-(0.00)	-(0.01)	-(0.31)	-(0.05)	-(0.01)	-(0.01)
200	1000	Dense	2	1.13 (0.01)	7.93 (0.20)	2.51 (0.02)	49.75 (0.64)	0.90 (0.03)	-(0.01)	34.02 (0.02)
200	1000	Sparse	2	0.88 (0.00)	4.19 (0.09)	5.56 (0.10)	49.73 (0.64)	0.60 (0.04)	-(0.01)	7.70 (0.10)
200	1000	Mixed	2	0.94 (0.00)	6.62 (0.17)	4.05 (0.05)	50.51 (0.65)	0.98 (0.03)	-(0.01)	16.65 (0.05)
200	1000	Dense	5	0.61 (0.00)	9.81 (0.57)	3.51 (0.06)	54.88 (2.83)	1.54 (0.18)	-(0.01)	4.57 (0.06)
200	1000	Sparse	5	0.44 (0.00)	5.12 (0.25)	6.22 (0.24)	55.74 (2.74)	1.52 (0.18)	-(0.01)	6.15 (0.24)
200	1000	Mixed	5	0.50 (0.00)	7.50 (0.39)	4.59 (0.15)	57.64 (2.82)	1.47 (0.21)	-(0.01)	5.13 (0.15)
Average				0.65 (0.01)	7.78 (0.34)	2.97 (0.08)	48.36 (1.82)	1.18 (0.11)	33.90 (1.33)	4.77 (0.08)

take the change-vector θ to satisfy $\theta \propto (I_1, \dots, I_k, 0, \dots, 0)^\top$, where I_1, \dots, I_k are drawn independently and uniformly from $\{-1, 1\}$. Furthermore, we scale θ such that $\Delta\varphi^2 = (n/5) \|\theta\|_2^2 = 9r(k)$.

Similar to the simulation study in [21], we consider the following data generating mechanisms. In model M_0 we take the noise vector W_v to satisfy $W_v \sim N_p(0, I)$ independently for $v \in [n]$. In models M_{Unif} and M_{t_d} we take $W_{i,v} \sim \text{Unif}(-\sqrt{3}, \sqrt{3})$ and $\{d/(d-2)\}^{1/2} W_{i,v} \sim t_d$, respectively and independently for all $v \in [n]$ and $i \in [p]$, where t_d denotes the Student t distribution with d degrees of freedom. In model $M_{\text{cs, loc}}(\rho)$ we let the noise vectors W_1, \dots, W_n have short-ranged spatial correlation, taking $W_v \sim N_p(0, \Sigma(\rho))$ independently for all $v \in [n]$, where $\Sigma(\rho)_{j,k} = \rho^{|j-k|}$ for each $j, k \in [p]$. In the model $M_{\text{cs}}(\rho)$ we let the noise vectors W_1, \dots, W_n have global spatial correlation by taking $W_v \sim N_p(0, \Delta(\rho))$ independently for $v \in [n]$, where $\Delta(\rho) = (1-\rho)I_p + \rho/p I_p I_p^\top$. In the model $M_{\text{temp}}(\rho)$ we allow for temporal dependence between the noise vectors W_1, \dots, W_n by letting $W_1 = \widetilde{W}_1$ and $W_v = \sqrt{\rho} \widetilde{W}_v + \sqrt{1-\rho} W_{v-1}$ for $v = 2, \dots, n$, where $\widetilde{W}_1, \dots, \widetilde{W}_n \sim N_p(0, I_p)$, independently. In the models M_{async} and M_{gradual} we allow for changes in the mean to occur asynchronous and gradual in time, respectively, with noise vectors $W_v \sim N_p(0, I_p)$ independently for $v \in [n]$. In M_{async} , for each changepoint η_j , we randomly shift the position (in time) of the change in mean in the i th coordinate, where the shifts are drawn independently from $\text{Unif}(\eta_j - \lfloor \Delta_j/2 \rfloor, \eta_j - \lfloor \Delta_j/2 \rfloor + 1, \dots, \eta_j + \lfloor \Delta_j/2 \rfloor)$. In M_{gradual} , for each changepoint η_j , any change in mean occurs linearly over time, starting at position $\eta_j - \lfloor \Delta_j/2 \rfloor + 1$ and ending at position $\eta_j + \lfloor \Delta_j/2 \rfloor + 1$.

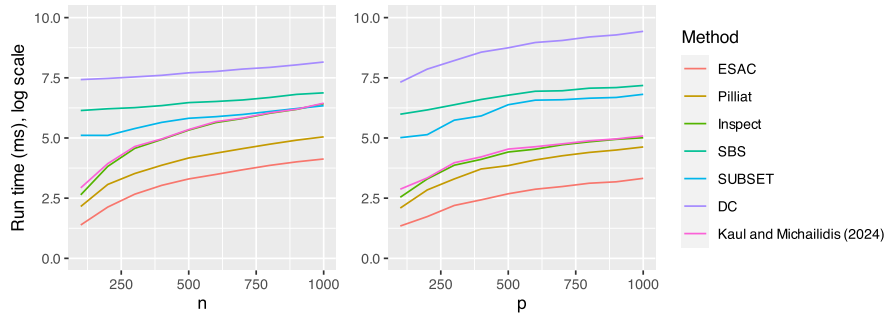


FIG 4. run times of the methods as functions of n (left) and p (right) on a logarithmic scale.

Table 4 displays the MSE of the competing methods using the same running parameters as in Section 4.1, based on $N = 1000$ runs. Table 4 indicates that ESAC (along with Inspect and SUBSET) is robust to model deviations in the form of light-tailed noise and short-ranged spatial correlation. With global spatial correlation, however, all methods degrade substantially in performance, with Inspect having a slight edge over the remaining methods. With auto-correlation, the performance of the methods also degrades markedly, with ESAC and SUBSET having a slight edge over the remaining methods. Lastly, ESAC, SUBSET and the method of [10] seem to be slightly more robust to asynchronous and gradual occurrence of changepoints than the remaining methods.

5. Real data example

To illustrate how ESAC can be applied in practice, we examine raw sensor data from a Swedish hydro power plant. The data consists of measurements from 20 sensors taken every minute for 1800 minutes, so that $p = 20$ and $n = 1800$. The sensors measure the magnitude of movements and vibrations (the latter measured at 1–10 and 10–1000 Hz bands) at various locations along the shaft connecting the turbine and the generator. During the 1800 minutes we consider, the mode of operation changes several times, detailed in Table 5. We take these changes of operation mode as the ground truth regarding the number of changepoints and their locations.

The data generating mechanism of the data is undeniably in violation of several underlying assumptions of ESAC. Importantly, the data are highly cross-correlated and auto-regressive. Moreover, the measurements in the data set are influenced by contextual variables such as power output, guide vane opening, and other (human controlled) running conditions in a complex manner. This dependence on contextual variables should ideally be modeled carefully, although such modeling is outside the scope of this paper. As a remedy, we instead transform the observed data by right multiplying each observed data point X_i by $\widehat{\Sigma}^{-1/2}$. Here, $\widehat{\Sigma}$ is the estimated variance-covariance matrix of X_i , estimated

TABLE 4
Single changepoint estimation under misspecified model.

Parameters		MSE					
Model	Sparsity	ESAC	Inspect	SBS	SUBSET	DC	Kaul et. al 2021
M	Sparse	1.2	1.0	506.1	1.1	25.4	2.1
M	Dense	1.6	84.5	1507.9	1.6	762.6	4206.8
M_{Unif}	Sparse	1.0	1.2	666.7	0.9	17.7	15.3
M_{Unif}	Dense	7.5	141.3	1506.1	23.4	977.6	6594.8
M_{t_3}	Sparse	1373.5	405.7	3038.7	1379.4	12.7	181.2
M_{t_3}	Dense	1561.6	1184.0	12187.7	1642.9	1368.1	1035.8
$M_{t_{10}}$	Sparse	2.3	1.5	649.5	1.4	32.6	0.7
$M_{t_{10}}$	Dense	2.1	54.6	2045.5	2.1	967.6	2745.3
$M_{\text{cs, loc}}(\rho = 0.1)$	Sparse	1.1	0.9	479.2	1.0	14.6	1.0
$M_{\text{cs, loc}}(\rho = 0.1)$	Dense	2.1	98.7	1517.5	2.1	760.1	3860.5
$M_{\text{cs, loc}}(\rho = 0.4)$	Sparse	1.4	0.9	501.0	1.0	23.4	0.6
$M_{\text{cs, loc}}(\rho = 0.4)$	Dense	4.8	130.8	1508.9	4.8	1016.8	4414.8
$M_{\text{cs}}(\rho = 0.1)$	Sparse	169.5	3.2	490.3	74.2	14.6	20.1
$M_{\text{cs}}(\rho = 0.1)$	Dense	297.7	167.2	1509.0	297.7	953.1	4377.3
$M_{\text{cs}}(\rho = 0.4)$	Sparse	3978.0	94.8	520.3	3918.1	64.8	192.3
$M_{\text{cs}}(\rho = 0.4)$	Dense	5384.2	1383.5	1519.5	5387.8	2142.3	5779.7
$M_{\text{AR}}(\rho = 0.1)$	Sparse	148.0	77.1	193.9	148.0	189.9	1676.3
$M_{\text{AR}}(\rho = 0.1)$	Dense	40.6	461.7	3023.6	40.6	2395.2	1511.7
$M_{\text{AR}}(\rho = 0.4)$	Sparse	979.9	1648.6	1209.3	979.9	1968.5	1442.0
$M_{\text{AR}}(\rho = 0.4)$	Dense	1274.3	1994.0	2556.8	1274.3	4406.8	1641.4
$M_{\text{asyn}}(\rho = 0.1)$	Sparse	83.1	99.0	603.8	81.2	212.5	43.9
$M_{\text{asyn}}(\rho = 0.1)$	Dense	75.8	288.3	1521.7	79.4	1464.4	4707.8
$M_{\text{grad}}(\rho = 0.1)$	Sparse	50.3	54.5	794.5	49.2	162.6	49.4
$M_{\text{grad}}(\rho = 0.1)$	Dense	67.7	231.3	1524.4	92.2	1762.9	4527.0
Average MSE		646.2	358.7	1732.6	645.2	904.9	2042.8

TABLE 5
Operation modes of the hydro power plant.

Time period	Operation mode
1–529	running
530–537	stopping
538–1307	off
1308–1310	starting
1311–2000	running

from an independent data set with 5992 observations, in which running conditions are stable (i.e. with no changes in operation mode). Moreover, we choose the penalty function $\gamma(t)$ empirically as described in Appendix B, using false probability rate $\varepsilon = 0.01$ and letting each of the $N = 1000$ Monte Carlo samples $X^{(j)}$ have independent entries following a t_5 distribution. This choice of penalty function ensures that ESAC is rather conservative in declaring changepoints.

The Monte Carlo simulation for generating the penalty function $\lambda(t)$ took 2 minutes and 2 seconds. Applying ESAC to the data took 0.035 seconds, resulting in six estimated changepoints, at locations 531, 533, 974, 1067, 1308, and 1330. Figure 5 displays the 20 transformed sensor measurements over the sampling period, with estimated changepoint locations indicated by red ticks on the x axis. The gray rectangle in the plot indicate the times at which the

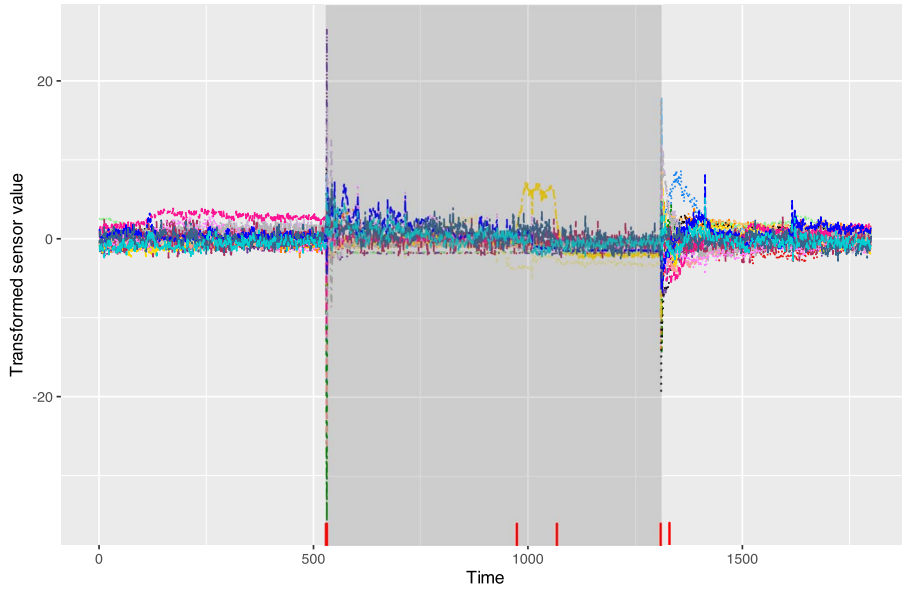


FIG 5. Transformed sensor measurements with detected changepoints indicated by red ticks. Grey areas indicate time intervals in which the plant is starting, stopping or off.

plant is either starting, stopping, or off. From the Figure, we clearly see that the first, second and fifth and sixth identified changepoint are associated with a change in operation mode of the plant. Interestingly, the other two changepoints, located at time points 974, 1067, are not associated with a change in running conditions. These changepoints are likely declared by ESAC due to the sudden shift in the yellow curves occurring at time 974 and reverting back again at time 1067.

Appendix A: Proofs of main results

Proof of Proposition 3.1. Set $c_1 = 6 + 2 \log(8/\varepsilon)/\log(2)$, $c_2 = 12 + 2(\log(1/\varepsilon))^{1/2} + 2 \log(1/\varepsilon)$ and $\gamma_0 = 9(c_1 + c_1^{1/2} \exp(-1)) + c_2$. Note first that I has cardinality no larger than n^3 . By a union bound, it thus suffices to show that $\mathbb{P}(S_{\gamma, (s,e]} > 0) \leq \varepsilon n^{-3}$ for any $(s, e) \subseteq I$.

So fix any $(s, e) \subseteq I$. Let $t \in \mathcal{T} \setminus \{p\}$ (the case $t = p$ is handled later), and fix $x_t > 0$ (to be specified shortly). Since (s, e) does not contain any changepoint, we must have $C_{(s,e]}^v(i) \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$ for all $i \in [p]$. By Lemma F.2 we have that

$$\sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(t)} \right\} \mathbb{1} \left\{ \left| C_{(s,e]}^v(i) \right| > a(t) \right\} \geq 9 \left[\left\{ p e^{-a(t)^2/2} x_t \right\}^{1/2} + x_t \right]$$

with probability at most e^{-x_t} . Now set $x_t = c_1 \left\{ \frac{p \log^2(n)}{t^2} \wedge r(t) \right\}$ for all t . Then,

$$\sum_{t \in \mathcal{T} \setminus \{p\}} e^{-x_t} \leq \sum_{t \in \mathcal{T} \setminus \{p\}} \exp \left\{ -c_1 \frac{p \log^2(n)}{t^2} \right\} + \sum_{t \in \mathcal{T} \setminus \{p\}} \exp \{-c_1 r(t)\}.$$

For the first sum, we have

$$\begin{aligned} \sum_{t \in \mathcal{T} \setminus \{p\}} \exp \left\{ -c_1 \frac{p \log^2(n)}{t^2} \right\} &\leq \sum_{k=0}^{\infty} \exp \{-c_1 \log(n) 4^k\} \\ &= \sum_{k=0}^{\infty} \left(\frac{1}{n^{c_1}} \right)^{4^k} \\ &\leq n^{-c_1} + n^{-c_1} \sum_{k=1}^{\infty} \left(\frac{1}{n^{c_1}} \right)^{3k} \\ &= 2n^{-c_1}. \end{aligned}$$

For the second sum, noting that $c_1 r(t) = c_1 \left\{ t \log \left(\frac{ep \log n}{t^2} \right) \vee \log n \right\} \geq (c_1/2)t \log \left(\frac{ep \log n}{t^2} \right) + (c_1/2) \log n$, we have

$$\begin{aligned} \sum_{t \in \mathcal{T} \setminus \{p\}} \exp \{-c_1 r(t)\} &\leq n^{-c_1/2} \exp(-c_1/2) \sum_{t \in \mathcal{T} \setminus \{p\}} \left(\frac{t^2}{ep \log n} \right)^{c_1 t/2} \\ &\leq n^{-c_1/2} \exp(-c_1/2) \left(1 + \sum_{k=1}^{\infty} 4^{-c_1 k/2} \right) \\ &\leq 2n^{-c_1/2}. \end{aligned}$$

With this choice of x_t , we thus have that

$$\sum_{t \in \mathcal{T} \setminus \{p\}} e^{-x_t} \leq 4n^{-c_1/2},$$

using that $c_1 > 1$. Moreover, using that $a^2(t) = 4 \log \left(\frac{ep \log n}{t^2} \right)$, we have that

$$\begin{aligned} 9 \left[\left\{ p e^{-a^2(t)/2} x_t \right\}^{1/2} + x_t \right] &= 9 \left[\left\{ p \frac{t^4}{e^2 p^2 \log^2 n} x_t \right\}^{1/2} + x_t \right] \\ &\leq 9 \left\{ \frac{t c_1^{1/2}}{e} + c_1 r(t) \right\} \\ &\leq 9 \left(c_1^{1/2} \exp(-1) + c_1 \right) r(t), \end{aligned}$$

where we used that $x_t \leq c_1 r(t)$ and $x_t \leq c_1 p \log^2(n)/t^2$, as well as the fact that $t \leq r(t)$ whenever $t \leq (p \log n)^{1/2}$. Recalling that $\gamma(t) = \gamma_0 r(t)$, since $\gamma_0 > 9 \left(c_1 + c_1^{1/2} \exp(-1) \right)$, a union bound gives

$$\mathbb{P}(\exists t \in \mathcal{T} \setminus \{p\} ; S_{\gamma, (s, e]}(t) \geq 0)$$

$$\begin{aligned}
 &= \mathbb{P} \left[\exists t \in \mathcal{T} \setminus \{p\} ; \sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(t)} \right\} \mathbb{1} \left\{ |C_{(s,e]}^v(i)| > a(t) \right\} \geq \gamma_0 r(t) \right] \\
 &\leq 4n^{-c_1/2} \\
 &\leq 4n^{-3 - \log(8/\varepsilon)/\log(2)} \\
 &\leq 4n^{-3} \exp(-\log(8/\varepsilon) \log(n)/\log(2)) \\
 &\leq n^{-3} \varepsilon/2,
 \end{aligned}$$

where we in the last inequality used that $n \geq 2$.

Now consider the case where $t = p$. If $p \leq (p \log n)^{1/2}$, then similarly as above we have that

$$\mathbb{P} \left(S_{\gamma, (s,e]}(p) \geq 0 \right) \leq n^{-3} \varepsilon/2.$$

If we instead have $p > (p \log n)^{1/2}$ (in which case $a(p) = 0$ and $\nu_{a(p)} = 1$), then

$$\sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(p)} \right\} \mathbb{1} \left\{ |C_{(s,e]}^v(i)| > a(p) \right\} = \sum_{i=1}^p C_{(s,e]}^v(i)^2 - p.$$

As $\sum_{i=1}^p C_{(s,e]}^v(i)^2 \sim \chi_p^2$, we obtain from Lemma F.4 that

$$\mathbb{P} \left\{ \sum_{i=1}^p C_{(s,e]}^v(i)^2 - p > 2(p \log(2n^3/\varepsilon))^{1/2} + 2 \log(2n^3/\varepsilon) \right\} \leq n^{-3} \varepsilon/2.$$

Now,

$$\begin{aligned}
 &2(p \log(2n^3/\varepsilon))^{1/2} + 2 \log(2n^3/\varepsilon) \\
 &\leq 2(p \log(n^4/\varepsilon))^{1/2} + 2 \log(n^4/\varepsilon) \\
 &\leq 4(p \log n)^{1/2} + 2(p \log(1/\varepsilon))^{1/2} + 8 \log n - 2 \log(\varepsilon) \\
 &\leq r(p) \left(12 + 2(\log(1/\varepsilon))^{1/2} + 2 \log(1/\varepsilon) \right) \\
 &= c_2 r(p) \\
 &< \gamma_0 r(p),
 \end{aligned}$$

using that $n \geq 2$, $r(p) \geq 1$, and $r(p) = (p \log n)^{1/2} \geq \log n$ whenever $p \geq (p \log n)^{1/2}$. Hence,

$$\mathbb{P} \left\{ \sum_{i=1}^p C_{(s,e]}^v(i)^2 - p > \gamma_0 r(p) \right\} \leq n^{-3} \varepsilon/2.$$

We conclude that

$$\mathbb{P} \left(\max_{(s,e] \in I} S_{\gamma, (s,e]} > 0 \right) \leq n^3 \mathbb{P} \left(\exists t \in \mathcal{T} ; S_{\gamma, (s,e]}(t) \geq 0 \right)$$

$$\begin{aligned} &\leq n^3 (n^{-3}\varepsilon/2 + n^{-3}\varepsilon/2) \\ &= \varepsilon, \end{aligned}$$

and we are done. □

Proof of Theorem 3.2. Let $\lambda_0 \geq 63$, $\lambda(t) = \lambda_0 r(t)$, $C_1 = 2\{2(4\lambda_0 + 242)\}^{1/2} + 2\lambda_0 + 123$ and $C_0 > C_1$. Let $\hat{\eta} \in \arg \max_{0 < v < n} S_\lambda^v$, where S_λ^v is defined as in (6), and let $S^v(t)$ be defined as in (3). Let the CUSUM transformation $T_{(s,e]}^v(\cdot)$ be defined as in (2), and for ease of notation, let $T^v(\cdot) = T_{(0,n]}^v(\cdot)$. Let $\mathcal{K} = \{i ; \mu_{i,\eta+1} - \mu_{i,\eta} \neq 0\}$ denote the set of coordinates for which there is a change in mean, and for any $0 < v < n$ let $\beta_v = \sum_{i \in \mathcal{K}} \{T^\eta(\mu_{i,\cdot})^2 - T^v(\mu_{i,\cdot})^2\}$. Let \bar{k} denote the smallest element in \mathcal{T} such that $\bar{k} \geq k$. We may without loss of generality take $\sigma = 1$, as we can otherwise normalize the data matrix X and replace the squared norm of the change in mean φ^2 by φ^2/σ^2 .

Consider the event $\mathcal{E} = \mathcal{E}_1 \cap \mathcal{E}_2 \cap \mathcal{E}_3 \cap \mathcal{E}_4$ as defined in Lemma F.5, for which we know that $\mathbb{P}(\mathcal{E}) \geq 1 - \frac{1}{n}$. On the event \mathcal{E} , we will show that any $0 < v < n$ such that $|v - \eta| > C_1 h(k)/\varphi^2$ must satisfy $S_\lambda^\eta > S_\lambda^v$, which implies that $|\hat{\eta}_\lambda - \eta| \leq C_1 h(k)/\varphi^2$.

Fix some $0 < v < n$ and let $t^* \in \arg \max_{t \in \mathcal{T}} S_\lambda^v(t)$, so that $S_\lambda^v = S_\lambda^v(t^*)$. We claim that

$$S_\lambda^\eta - S_\lambda^v \geq \beta_v - 2\{2\beta_v h(k)\}^{1/2} - (119 + 2\lambda_0)h(k). \tag{13}$$

To see this, suppose first that $k \leq (p \log n)^{1/2}$. We have that

$$\begin{aligned} S_\lambda^\eta - S_\lambda^v &\geq S_\lambda^\eta(\bar{k}) - S_\lambda^v(t^*) \\ &= \sum_{i=1}^p \left[\left(C^\eta(i)^2 - \nu_{a(\bar{k})} \right) \mathbb{1}\{|C^\eta(i)| > a(\bar{k})\} \right. \\ &\quad \left. - \left(C^v(i)^2 - \nu_{a(t^*)} \right) \mathbb{1}\{|C^v(i)| > a(t^*)\} \right] \\ &\quad - \lambda_0 r(\bar{k}) + \lambda_0 r(t^*). \end{aligned}$$

For any $x \in \mathbb{R}$ and any $t \in \mathcal{T}$, we have $x^2 - \nu_{a(t)} \leq (x^2 - \nu_{a(t)}) \mathbb{1}\{|x| > a(t)\} \leq x^2$, and so

$$\begin{aligned} S_\lambda^\eta - S_\lambda^v &\geq \sum_{i \in \mathcal{K}} [C^\eta(i)^2 - C^v(i)^2] - k\nu_{a(\bar{k})} \\ &\quad + \sum_{i \in [p] \setminus \mathcal{K}} \left[\left(C^\eta(i)^2 - \nu_{a(\bar{k})} \right) \mathbb{1}\{|C^\eta(i)| > a(\bar{k})\} \right. \\ &\quad \left. - \left(C^v(i)^2 - \nu_{a(t^*)} \right) \mathbb{1}\{|C^v(i)| > a(t^*)\} \right] \\ &\quad - \lambda_0 r(\bar{k}) + \lambda_0 r(t^*). \end{aligned}$$

On the event $\mathcal{E}_1 \cap \mathcal{E}_2 \cap \mathcal{E}_3 \supseteq \mathcal{E}$ we therefore have that

$$S_\lambda^\eta - S_\lambda^v \geq \beta_v - 2\{2\beta_v \log n\}^{1/2} - 16r(k) - 35r(\bar{k})$$

$$\begin{aligned} & - (63 - \lambda_0)r(t^*) - \lambda_0 r(\bar{k}) - k\nu_{a(\bar{k})} \\ & \geq \beta_v - 2 \{2\beta_v r(k)\}^{1/2} - (92 + 2\lambda_0)r(k), \end{aligned}$$

where we have used that $\lambda_0 \geq 63$, $\log n \leq r(k)$, $r(\bar{k}) \leq 2r(k)$, and for $k \leq (p \log n)^{1/2}$, we have $k\nu_{a(\bar{k})} \leq k(2 + a^2(\bar{k})) \leq 2k + ka^2(k) \leq 6r(k)$. Since $r(k) \leq h(k)$ for all $k \in [p]$, the claim (13) holds whenever $k \leq (p \log n)^{1/2}$.

Now suppose $k > (p \log n)^{1/2}$. By the definition of \mathcal{E}_4 , we have that $S_\lambda^\eta(t^*) - S_\lambda^v(p) \leq 5h(p) + 63r(t^*) - \lambda_0 r(t^*) + \lambda_0 r(p)$. Hence,

$$\begin{aligned} S_\lambda^\eta - S_\lambda^v & \geq S_\lambda^\eta(p) - S_\lambda^v(t^*) \\ & = S_\lambda^\eta(p) - S_\lambda^v(p) + S_\lambda^v(p) - S_\lambda^v(t^*) \\ & \geq S_\lambda^\eta(p) - S_\lambda^v(p) - 5h(p) - 63r(t^*) + \lambda_0 r(t^*) - \lambda_0 r(p) \\ & = \sum_{i=1}^p \{C^\eta(i)^2 - C^v(i)^2\} - 5h(p) - 63r(t^*) + \lambda_0 r(t^*) - \lambda_0 r(p). \end{aligned}$$

On the event \mathcal{E} we thus have that

$$\begin{aligned} S_\lambda^\eta - S_\lambda^v & \geq \beta_v - 2 \{2\beta_v \log n\}^{1/2} - 16r(p) - 63r(p) - 35r(p) - 5h(p) \\ & \quad - 63r(t^*) + \lambda_0 r(t^*) - \lambda_0 r(p) \\ & \geq \beta_v - 2 \{\beta_v h(p)\}^{1/2} - (119 + \lambda_0)h(p), \end{aligned}$$

where we in the last inequality used that $r(k) \leq h(k)$ for all k and $\lambda_0 \geq 63$. Hence (13) holds whenever $k > (p \log n)^{1/2}$. Solving the quadratic inequality (13) with respect to β_v , we obtain that $S_\lambda^\eta - S_\lambda^v > 0$ if

$$\beta_v > \left\{ 2(4\lambda_0 + 242)^{1/2} + 2\lambda_0 + 123 \right\} h(k). \quad (14)$$

Without loss of generality we may assume $v \geq \eta$ (the converse case is similar). By Lemma F.11 we have that

$$\begin{aligned} \beta_v & = \sum_{i \in \mathcal{K}} \{T^\eta(\mu_{i,\cdot})^2 - T^v(\mu_{i,\cdot})^2\} \\ & = \frac{|v - \eta|\eta}{|v - \eta| + \eta} \varphi^2 \\ & \geq \frac{1}{2} \min(|v - \eta|, \eta) \varphi^2, \end{aligned}$$

and therefore (14) is satisfied if

$$\begin{aligned} \min(|v - \eta|, \eta) & > 2 \left\{ 2(4\lambda_0 + 242)^{1/2} + 2\lambda_0 + 123 \right\} \frac{h(k)}{\varphi^2} \\ & = C_1 \frac{h(k)}{\varphi^2}. \end{aligned} \quad (15)$$

By the assumption $C_0 > C_1$, η is strictly larger than the right hand side of (15). Therefore (14) is satisfied if

$$|v - \eta| > C_1 \frac{h(k)}{\varphi^2}.$$

Hence, if $|\eta - v| > C_1 h(k)/\varphi^2$, we must have $S_\lambda^\eta > S_\lambda^v$, and the proof is complete. \square

Proof of Theorem 3.3. Let $\gamma_0 \geq 82$, $\gamma(t) = \gamma_0 r(t)$ and define $C_1 = 32\{\gamma_0 + 170 + 8(2\gamma_0 + 276)^{1/2}\}$ and $C_0 = 2C_1$. We may without loss of generality take $\sigma = 1$, as we can otherwise normalize the data matrix X and replace the squared norm of the change in mean φ^2 by φ^2/σ^2 . Let $\mathcal{M} = \{(s_m, e_m) ; m \in [M]\}$ denote the (enumerated) collection of seeded intervals generated by Algorithm 4. In the following, we will use the name ESAC to refer to either Algorithm 2 or 3. We work on the event $\mathcal{E} = \mathcal{E}_5 \cap \mathcal{E}_6$ as defined in Lemma F.6, for which we know that $\mathbb{P}(\mathcal{E}) \geq 1 - 1/n$. The proof goes as follows. In step 1 we show that each changepoint η_j will be detected using a seeded interval with certain properties. In step 2, by an inductive argument, we show that ESAC detects all changepoints within the given error-rate.

Step 1. We first claim that, for each j in $[J]$, there exists a seeded interval $(s_{\bar{m}}, e_{\bar{m}}] = (v - l, v + l] \in \mathcal{M}$ such that the following holds

- (P1) $C_1 r(k_j)/(4\varphi_j^2) \leq l \leq C_1 r(k_j)/(2\varphi_j^2) \vee 1$;
- (P2) $|\eta_j - v| \leq l/2$;
- (P3) $s_{\bar{m}} \geq \eta_j - (\Delta_j/2 \vee 1)$;
- (P4) $e_{\bar{m}} \leq \eta_j + (\Delta_j/2 \vee 1)$;
- (P5) $S_{\gamma, (s_{\bar{m}}, e_{\bar{m}}]}^v \geq 0$.

To see this, fix any $j \in [J]$, and let $h = C_1 r(k_j)/(2\varphi_j^2)$. Now let $(s_{\bar{m}}, e_{\bar{m}}] = (v - l, v + l]$ denote the seeded interval from Lemma F.7. Then properties (P1) and (P2) follow immediately. Moreover, as $\varphi_j^2 \Delta_j \geq C_0 r(k_j)$ (by the SNR assumption (12) in Theorem 3.3) and $C_0 \geq 2C_1$, we have $h \leq \Delta_j/4$. The properties (P3) and (P4) then follow from Lemma F.7. To show the last property (P5), observe first that

$$S_{\gamma, (s_{\bar{m}}, e_{\bar{m}}]}^v \geq \beta_{(s_{\bar{m}}, e_{\bar{m}}]}^v - 8 \left\{ 2\beta_{(s_{\bar{m}}, e_{\bar{m}}]}^v r(k_j) \right\}^{1/2} - (\gamma_0 + 106) r(k_j),$$

on the event \mathcal{E} , where $\beta_{(s_{\bar{m}}, e_{\bar{m}}]}^v = \sum_{i=1}^P T_{(s_{\bar{m}}, e_{\bar{m}}]}^v(\mu_{i,\cdot})^2$. By solving the quadratic inequality, we obtain that $S_{\gamma, (s_{\bar{m}}, e_{\bar{m}}]}^v \geq 0$ whenever

$$\begin{aligned} \beta_{(s_{\bar{m}}, e_{\bar{m}}]}^v &\geq \left\{ \gamma_0 + 170 + 8(2\gamma_0 + 276)^{1/2} \right\} r(k_j) \\ &= C_1/32 r(k_j). \end{aligned}$$

Assume without loss of generality that $\eta_j \leq v$ (the converse case is similar). By the definition of the CUSUM, and using that $|\eta_j - v| \leq l/2$, we get that

$$\beta_{(s_{\bar{m}}, e_{\bar{m}}]}^v = \frac{v - s_{\bar{m}}}{(e_{\bar{m}} - s)(e_{\bar{m}} - v)} (e_{\bar{m}} - \eta_j)^2 \varphi_j^2$$

$$\begin{aligned} &\geq \frac{1}{2l}(l/2)^2\varphi_j^2 \\ &= l\varphi_j^2/8. \end{aligned}$$

Since $l \geq C_1r(k_j)/(4\varphi_j^2)$, we must have that $\beta_{(s_{\bar{m}}, e_{\bar{m}}]}^v \geq C_1/32r(k_j)$, which implies (P5).

Step 2. We continue the proof as follows. By induction, with some slight abuse of notation, it suffices to consider any integer interval $(s, e] \subseteq (0, n]$ such that

$$\eta_{h-1} \leq s < \eta_h < \dots < \eta_{h+q} < e \leq \eta_{q+h+1},$$

for some $q \geq -1$, and, whenever $q > -1$,

$$\begin{aligned} s &\leq \eta_h - \Delta_h/2; \\ e &\geq \eta_{h+q} + \Delta_{h+q}/2. \end{aligned}$$

Note that $q = -1$ corresponds to there being no changepoint in the open integer interval (s, e) . We consider this case first. For any seeded interval $(s_m, e_m] \subseteq (s, e]$ and any $s_m < v < e_m$, we will have that $S_{\gamma, (s_m, e_m]}^v < 0$, due to the definition of the event \mathcal{E} . Hence no changepoint will be declared by ESAC in this case.

Now consider the case where $q > -1$. Note first that a changepoint will be declared by the ESAC algorithm. Indeed, for the h th changepoint we may take $(s_{\bar{m}}, e_{\bar{m}}]$ as in step 1, for which the properties (P3) and (P4) imply that $(s_{\bar{m}}, e_{\bar{m}}] \subseteq (s, e]$, due to the inductive hypothesis. By property (P5), we know that $S_{\gamma, (s_{\bar{m}}, e_{\bar{m}}]}^v \geq 0$, and hence a changepoint will be detected in (s, e) . This implies that $\mathcal{O}_{(s, e]}$, as defined in the ESAC algorithm, satisfies $\mathcal{O}_{(s, e]} \neq \emptyset$. Now let m^* , v^* and l^* be as defined in the ESAC algorithm. Note that (s_{m^*}, e_{m^*}) must contain a changepoint, say η_j , as we by the definition of \mathcal{E} otherwise would have had $S_{\gamma, (s_{m^*}, e_{m^*})}^v < 0$ for any $s_{m^*} < v < e_{m^*}$. Further, since ESAC uses the narrowest possible seeded interval to estimate a changepoint, we must have that $l^* = (e_{m^*} - s_{m^*})$ satisfies $l^* \leq e_{\bar{m}} - s_{\bar{m}} \leq C_1r(k_j)/(\varphi_j^2) \vee 2$, where $(s_{\bar{m}}, e_{\bar{m}}]$ is the seeded interval as in the claim for η_j . Since $s_{m^*} < v^* < e_{m^*}$, it then follows that

$$|v^* - \eta_j| \leq \{C_1r(k_j)/\varphi_j^2 \vee 2\} - 2 \leq C_1 \frac{r(k_j)}{\varphi_j^2}.$$

It remains to show that the two new segments in the recursive step, $(s, s_{m^*} + 1]$ and $(e_{m^*} - 1, e]$ satisfy the inductive hypothesis. Without loss of generality consider $(s, s_{m^*} + 1]$ (the argument for the other interval is similar), and suppose that $j \geq h + 1$ (otherwise there is nothing to show). To show that the inductive hypothesis holds for $(s, s_{m^*} + 1]$, it suffices to show that $s_{m^*} + 1 \geq \eta_{j-1} + \Delta_{j-1}/2$. As $\eta_j \in (s_{m^*}, e_{m^*})$, we must have $e_{m^*} \geq \eta_j + 1$. Hence

$$s_{m^*} + 1 = e_{m^*} + 1 - l^*$$

$$\begin{aligned}
&\geq \eta_j + 2 - \{C_1 r(k_j)/\varphi_j^2 \vee 2\} \\
&= \eta_{j-1} + (\eta_j - \eta_{j-1}) - \{C_1 r(k_j)/\varphi_j^2 - 2 \vee 0\} \\
&\geq \eta_{j-1} + (\eta_j - \eta_{j-1}) - \Delta_j/2 \\
&\geq \eta_{j-1} + (\eta_j - \eta_{j-1})/2 \\
&\geq \eta_{j-1} + \Delta_{j-1}/2,
\end{aligned}$$

where we in the first inequality used that $l^* = (e_{m^*} - s_{m^*}) \leq C_1 r(k_j)/\varphi_j^2 \vee 2$ and in the second inequality used that the SNR condition (12) implies $C_1 r(k_j)/\varphi_j^2 \leq \Delta_j/2$. Hence the inductive hypothesis holds for $(s, s_{m^*} + 1]$. \square

Proof of Proposition 3.4. Let \mathcal{M} denote the set of seeded intervals generated from Algorithm 4. Note first that computing and storing the cumulative sum of all rows of X requires $\mathcal{O}(np)$ FLOPs. Once these are stored, the number of FLOPs required to compute $C_{(s,e]}^v(j)$ as in (2) for some $(s, e] \in \mathcal{M}$ and some $s < v < e$ is of order $\mathcal{O}(p)$. Hence, the number of FLOPs required to compute $S_{\lambda, (s,e]}^v$ is of order $p|\mathcal{T}| = \mathcal{O}\{p \log(p \log n)\}$. In the best case there are $n - 1$ changepoints detected by the ESAC algorithm using all $n - 1$ intervals $(s, e] \in \mathcal{M}$ such that $e - s = 2$. In this case, the total number of FLOPs executed before ESAC terminates is of order $\mathcal{O}\{np + np \log(p \log n)\} = \mathcal{O}\{np \log(p \log n)\}$. In the worst case there are no changepoints detected by ESAC, in which case $S_{\lambda, (s,e]}^v$ has to be computed over each triple of integers s, v, e such that $s < v < e$ and $(s, e] \in \mathcal{M}$. By Lemma F.9, there are at most $\mathcal{O}(n \log n)$ distinct such triples. Hence the number of FLOPs executed before ESAC terminates in this case is of order $\mathcal{O}\{np \log n \log(p \log n)\}$. \square

Appendix B: Implementation details

To apply ESAC in practice, a choice must be made regarding the penalty functions λ, γ , estimation of σ , as well as the parameters α and K controlling the generation of seeded intervals. In this subsection we discuss these issues in turn, but first, we define two variants of ESAC as well as our algorithm for generating seeded intervals.

B.1. Slight modifications to ESAC

Algorithm 2 is a variant of the ESAC algorithm which features interval trimming. Here, the recursive step in the algorithm (the third and second last lines) differ from those found in 1. When Algorithm 2 declares a changepoint at location v^* , detected in the interval (s^*, e^*) , the remaining elements in interval (s^*, e^*) are never again used to detect or estimate changepoints.

A faster variant of Algorithm 2 is given by Algorithm 3. Algorithm 3 reduces the execution time by modifying step 3 in Algorithm 2 to only evaluate $S_{\gamma, (s_m, e_m]}^v$ at the mid-point $v_m = (s_m + e_m)/2$ of any seeded interval. Interestingly, the theoretical guarantees given by Theorem 3.3 also hold for this variant of ESAC,

unlike Algorithm 1. Note that the same modification can naturally be made to Algorithm 1 as well. In practice, we have experienced that Algorithm 3 has much lower power for detecting changepoints compared to 1. We therefore only recommend using Algorithm 3 when the consequent reduction in computational cost is necessary.

Algorithm 2 ESAC'(X, (s, e], M, B, γ, λ).

Input: Matrix of observations $X \in \mathbb{R}^{p \times n}$, left open and right closed integer interval $(s, e]$ in which candidate changepoints are searched for, an enumerated collection $\mathcal{M} = \{(s_m, e_m] ; m \in [M]\}$ of M half open integer sub intervals of $\{0, \dots, n\}$, a set of already detected changepoints \mathcal{B} , and penalty functions $\gamma(t), \lambda(t)$.
Output: Set \mathcal{B} of detected changepoints.

```

if  $e - s \leq 1$ :
    stop
set  $\mathcal{M}_{(s,e]} = \{m \in [M] : (s_m, e_m] \subset (s, e]\}$ 
set  $\mathcal{O}_{(s,e]} = \left\{ m \in \mathcal{M}_{(s,e]} : \max_{s_m < v < e_m} S_{\gamma, (s_m, e_m]}^v > 0 \right\}$ 
if  $\mathcal{O}_{(s,e]} = \emptyset$ 
    stop
set  $l^* = \min_{m \in \mathcal{O}_{(s,e]}} |e_m - s_m|$ 
set  $\mathcal{O}_{l^*} = \{m \in \mathcal{O}_{(s,e]} : |e_m - s_m| = l^*\}$ 
set  $m^* = \operatorname{argmax}_{m \in \mathcal{O}_{l^*}} \max_{s_m < v < e_m} S_{\lambda, (s_m, e_m]}^v$ 
set  $v^* = \operatorname{argmax}_{s_{m^*} < v < e_{m^*}} S_{\lambda, (s_{m^*}, e_{m^*})}^v$ 
 $\mathcal{B} \leftarrow \mathcal{B} \cup \{v^*\}$ 
 $\mathcal{B} \leftarrow \text{ESAC}'(X, (s, s_{m^*} + 1], \mathcal{M}, \mathcal{B}, \gamma, \lambda)$ 
 $\mathcal{B} \leftarrow \text{ESAC}'(X, (e_{m^*} - 1, e], \mathcal{M}, \mathcal{B}, \gamma, \lambda)$ 
return  $\mathcal{B}$ 

```

B.2. Efficient implementation of ESAC

The ESAC Algorithms 1, 2 and 3 are based on Narrowest-Over-Threshold selection of changepoints. Once a changepoint is detected in some seeded interval, say of length l^* , the changepoint location is estimated based only on intervals of length l^* . To minimize run time, any version of ESAC should therefore iterate through the seeded intervals $\{(s_m, e_m] : m \in [M]\}$ in the (increasing) order of their width. This computational trick gives significant speed improvements whenever changepoints can be detected by short seeded intervals.

B.3. Choice of α and K

The choice of α and K entails a trade-off between computational cost and statistical performance. As either α^{-1} or K increase, more seeded intervals are generated from Algorithm 4, increasing both the chance of detecting a changepoint and the run time of ESAC. After some experimentation, we have experienced that $\alpha = 3/2$ and $K = 4$ give a decent balance between run time and statistical accuracy.

Algorithm 3 ESAC'' ($X, (s, e], \mathcal{M}, \mathcal{B}, \gamma, \lambda$).

Input: Matrix of observations $X \in \mathbb{R}^{p \times n}$, left open and right closed integer interval $(s, e]$ in which candidate changepoints are searched for, an enumerated collection

$\mathcal{M} = \{(s_m, e_m] ; m \in [M]\}$ of M half open integer sub intervals of $\{0, \dots, n\}$, a set of already detected changepoints \mathcal{B} , and penalty parameters γ, λ .

Output: Set \mathcal{B} of detected changepoints.

```

if  $e - s \leq 1$ 
  stop
set  $\mathcal{M}_{(s,e]} = \{m \in [M] : (s_m, e_m] \subset (s, e]\}$ 
set  $v_m = \lfloor \frac{s_m + e_m}{2} \rfloor$  for all  $m = 1, \dots, M$ 
set  $\mathcal{O}_{(s,e]} = \{m \in \mathcal{M}_{(s,e]} : S_{\gamma, (s_m, e_m]}^{v_m} > 0\}$ 
if  $\mathcal{O}_{(s,e]} = \emptyset$ 
  stop
set  $l^* = \min_{m \in \mathcal{O}_{(s,e]}} |e_m - s_m|$ 
set  $\mathcal{O}_{l^*} = \{m \in \mathcal{O}_{(s,e]} : |e_m - s_m| = l^*\}$ 
set  $m^* = \operatorname{argmax}_{m \in \mathcal{O}_{l^*}} \max_{s_m < v < e_m} S_{\lambda, (s_m, e_m]}^v$ 
set  $v^* = \operatorname{argmax}_{s_{m^*} < v < e_{m^*}} S_{\lambda, (s_{m^*}, e_{m^*})}^v$ 
 $\mathcal{B} \leftarrow \mathcal{B} \cup \{v^*\}$ 
 $\mathcal{B} \leftarrow \text{ESAC}''(X, (s, s_{m^*} + 1], \mathcal{M}, \mathcal{B}, \gamma, \lambda)$ 
 $\mathcal{B} \leftarrow \text{ESAC}''(X, (e_{m^*} - 1, e], \mathcal{M}, \mathcal{B}, \gamma, \lambda)$ 
return  $\mathcal{B}$ 

```

B.4. Variance re-scaling

In the theoretical analysis of this paper, the noise level σ of each time series is assumed known and common across all p time series. In practice, this is an unrealistic assumption. As is common in the changepoint literature, we suggest estimating the noise level separately for each time series by the (scaled) Median Absolute Deviation (MAD) of first-order differences, as in e.g. [21]. If it is reasonable to assume that each time series has approximately the same noise level, the common noise level σ can be estimated for instance by taking a mean or median of the MAD estimates for each time series. Once estimates of the noise levels are obtained, the time series need only to be re-scaled by their estimated noise levels before applying ESAC.

B.5. Analytical choice of penalty functions

Recall that $\lambda(t)$ and $\gamma(t)$ are the penalty functions used in the sparsity-specific penalized score for changepoint localization and detection, respectively. The proofs of Theorems 3.2 and 3.3 provide suggestions for analytical choices of these penalty functions. However, we believe the leading constants are overly conservative. To obtain more practical choices of analytical penalizing functions, we have run simulations for combinations of n up to 1000 and p up to 5000. We have experienced that replacing n with n^4 in $a(t)$ (4) and $r(t)$ (8) gives a slightly better balance between the two terms $t \log(\frac{ep \log n}{t^2})$ and $\log n$ in $r(t)$.

As default values in our R package, as well as in the simulation study, we have replaced n with n^4 in $a(t)$ and $r(t)$. For changepoint estimation, we recommend using the penalty function

$$\tilde{\lambda}(t) = \begin{cases} \frac{3}{2} \left\{ (p \log n^4)^{1/2} + \log n^4 \right\} & \text{if } t \geq (p \log n)^{1/2}, \\ t \log \left(\frac{ep \log n^4}{t^2} \right) + \log n^4 & \text{otherwise.} \end{cases}$$

This recommendation is independent of whether each time series is re-scaled by Median Absolute Deviation (MAD) estimates. The choice of the two leading constants in $\tilde{\lambda}$ are the result of minimizing the Mean Squared Error (MSE) of the estimator (9) over a rough grid of n , p , η and k , where n has ranged from 200 to 1000 and p has ranged from 100 to 5000. For changepoint detection one can also use $\gamma(t) = \tilde{\lambda}(t)$, which in our experience gives a false positive rate of less than $1/n$. If the variance of each time series is re-scaled by MAD estimates, however, we recommend choosing the penalty function $\gamma(t)$ for changepoint detection using Monte Carlo simulation.

B.6. Empirical choice of penalty functions

To obtain exact control over the probability of a false changepoint being detected by ESAC, one can choose the penalty function $\gamma(t)$ by Monte Carlo simulation. Consider any false positive probability $\varepsilon > 0$ and Monte Carlo sample size N . A naive choice of empirical penalty function, denoted by $\hat{\gamma}_\varepsilon(t)$, is given by the following. Let \mathcal{M} denote the collection of seeded intervals to be used by ESAC. Simulate N data sets $(X^{(j)})_{j=1}^N$ following model (1) with no changepoints, in which each row is re-scaled by MAD estimates if applicable. If the data to be analyzed is expected to breach model assumptions, such as having heavy tailed noise, the $X^{(j)}$ can be simulated accordingly. For each $t \in \mathcal{T}$, let $\hat{\gamma}_\varepsilon(t)$ denote the $\lceil N(1 - \varepsilon) \rceil$ largest value of $\max_{(s,e] \in \mathcal{M}} \max_{s < v < e} S_{0,(s,e]}^v(X^{(j)})(t)$ over $j = 1, \dots, N$, where $S_{0,(s,e]}^v(X^{(j)})(t)$ is the sparsity-specific penalized score from (3) computed over the seeded interval $(s, e]$ with input matrix $X^{(j)}$ and with penalty function 0.

Due to multiple testing, the approximate false positive probability when using the naive penalty function $\hat{\gamma}_\varepsilon$ can only be upper bounded by $|\mathcal{T}|\varepsilon$. To adjust for multiple testing, a Bonferroni correction can easily be applied by replacing ε by $\varepsilon/|\mathcal{T}|$ in the definition of $\hat{\gamma}_\varepsilon(t)$. In our experience, though, such a Bonferroni correction is too conservative. An alternative approach to handle the multiple testing is to use the empirical penalty function $\hat{\gamma}^*(t) = r(t) \max_{s \in \mathcal{T}} \hat{\gamma}_\varepsilon(s)/r(s)$, in which the functional form is specified and only the leading constant is chosen by Monte Carlo simulation. In our experience, this approach also leads to an overly conservative penalty function, as the functional form of $\hat{\gamma}_\varepsilon(t)$ does not match the theoretical counterpart $r(t)$ exactly. We therefore recommend to use the following penalty function $\tilde{\gamma}(t)$, in which we introduce three separate leading constants for different segments of \mathcal{T} (and consequently a Bonferroni correction)

for slightly more flexibility. Let $\tilde{\gamma}(t)$ be defined by

$$\tilde{\gamma}(t) = \begin{cases} \tilde{\gamma}_1 r(t), & \text{for } t \leq \log n \wedge (p \log n)^{1/2} \\ \tilde{\gamma}_2 r(t), & \text{for } \log n < t \leq (p \log n)^{1/2} \\ \hat{\gamma}_{\varepsilon/3}(p), & \text{for } t = p, \end{cases}$$

where $\tilde{\gamma}_1$ and $\tilde{\gamma}_2$ are defined by

$$\tilde{\gamma}_1 = \max_{t \in \mathcal{T}; t \leq \log n} \hat{\gamma}_{\varepsilon/3}(t)/r(t),$$

$$\tilde{\gamma}_2 = \max_{t \in \mathcal{T}; \log n < t \leq \sqrt{(p \log n)}} \hat{\gamma}_{\varepsilon/3}(t)/r(t).$$

The penalty function $\tilde{\gamma}(t)$ ensures that the approximate probability of a false positive using ESAC is at most ε . We remark that the upper boundary of the first segment ($\log n$) is chosen somewhat ad hoc, while the second segment is the remaining region of the sparse regime, and the last segment is the dense regime. The empirical penalty function $\tilde{\gamma}(t)$ can also be used for changepoint estimation, i.e. setting $\tilde{\lambda}(t) = \tilde{\gamma}(t)$, although we have experienced that the analytical penalty function $\lambda(t)$ gives better performance in terms of MSE for Gaussian data.

B.7. Generation of seeded intervals

Given some sample size $n \geq 2$ and parameters $\alpha > 1$ and $K > 1$, Algorithm 4 generates a set of seeded intervals.

Algorithm 4 Seeded Interval Generation(α, K).

Input: Parameters α and K controlling the number of generated intervals

Output: Set of seeded intervals

```

Intervals  $\leftarrow$  {}
 $l \leftarrow 1$ 
while  $l \leq \frac{n}{2}$ :
  set  $s = \max \left\{ 1, \lfloor \frac{l}{K} \rfloor \right\}$ 
  for  $i = 0, \dots, \frac{n-2l}{s}$ :
    Intervals  $\leftarrow$  Intervals  $\cup \{(is, is + 2l)\}$ 
  Intervals  $\leftarrow$  Intervals  $\cup \{(n - 2l, n)\}$ 
   $l \leftarrow \max \{l + 1, \lfloor \alpha l \rfloor\}$ 
Return Intervals.

```

Appendix C: A narrowest-over-threshold variant of inspect

We have modified the Inspect algorithm [21], given by Algorithm 4. Instead of using Wild Binary Segmentation as search procedure, the methodology of Kovács et al. [12] is used. More specifically, the collection of integer sub-intervals is generated by Algorithm 4 instead of the random draws. Moreover, the location

of any detected changepoint is determined using only the narrowest intervals in which a changepoint is detected. We have given this modified version of Inspect the name NOTInspect, which is short for Narrowest-Over-Threshold Inspect. Formally, NOTInspect is defined as follows. For any $0 \leq s < e \leq n$, let $H^{(s,e]}$ denote the $p \times (e - s - 1)$ matrix in which the (i, j) th element is given by

$$H_{i,j}^{(s,e]} = T_{(s,e]}^{s+j}(X_{i,\cdot}),$$

i.e. the CUSUM of the i th row of X computed over the interval $(s, e]$ and evaluated at position $s + j$. For ease of notation, let $H_v^{(s,e]}$ denote the $(v - s)$ th column of $H^{(s,e]}$. Given $\lambda > 0$, let $\widehat{v}_\lambda^{(s,e]}$ denote the leading left singular vector of the matrix

$$\widehat{M}_\lambda = \arg \max_{M \in \mathcal{S}_2} \left(\left\langle H^{(s,e]}, M \right\rangle - \lambda \|M\|_1 \right),$$

where $\mathcal{S}_2 = \{M \in \mathbb{R}^{p \times (e-s-1)} : \|M\|_F \leq 1\}$.

Given an enumerated set $\mathcal{M} = \{(s_m, e_m)\}_{m=1}^M$ of M half open integer sub intervals of $0, \dots, n$, observations $X \in \mathbb{R}^{p \times n}$, and tuning parameters $\lambda, \xi > 0$, the NOTInspect algorithm is initiated by calling NOTInspect($X, (s, e], \mathcal{M}, \emptyset, \lambda, \xi$), and defined by Algorithm 5.

Algorithm 5 NOTInspect($X, (s, e], \mathcal{M}, \mathcal{B}, \lambda, \xi$).

Input: Matrix of observations $X \in \mathbb{R}^{p \times n}$, left open and right closed integer interval $(s, e]$ in which candidate changepoints are searched for, an enumerated collection $\mathcal{M} = \{(s_m, e_m) ; m \in [M]\}$ of M half open integer sub intervals of $\{0, \dots, n\}$, a set of already detected changepoints \mathcal{B} , and penalization parameters $\lambda, \xi > 0$.

Output: A set \mathcal{B} of detected changepoints.

```

if  $e - s \leq 1$ :
    stop
set  $\mathcal{M}_{(s,e]} = \{m : (s_m, e_m) \subset (s, e]\}$ 
set  $\mathcal{O}_{(s,e]} = \left\{ m \in \mathcal{M}_{(s,e]} : \max_{s_m < b < e_m} \left( \widehat{v}_\lambda^{(s_m, e_m]} \right)^\top H_{b_m}^{(s_m, e_m]} > \xi \right\}$ 
if  $\mathcal{O}_{(s,e]} = \emptyset$ :
    stop
set  $l^* = \min_{m \in \mathcal{O}_{(s,e]}} |e_m - s_m|$ 
set  $\mathcal{O}_{l^*} = \mathcal{O}_{(s,e]} \cap \{m : |e_m - s_m| = l^*\}$ 
set  $m^* = \operatorname{argmax}_{m \in \mathcal{O}_{l^*}} \max_{s_m < b < e_m} \left( \widehat{v}_\lambda^{(s_m, e_m]} \right)^\top H_b^{(s_m, e_m]}$ 
set  $b^* = \operatorname{argmax}_{s_{m^*} < b < e_{m^*}} \left( \widehat{v}_\lambda^{(s_{m^*}, e_{m^*}]} \right)^\top H_b^{(s_{m^*}, e_{m^*}]}$ 
 $\mathcal{B} \leftarrow \mathcal{B} \cup \{b^*\}$ 
 $\mathcal{B} \leftarrow \text{NOTInspect}(X, (s, b^*], \mathcal{M}, \mathcal{B}, \lambda, \xi)$ 
 $\mathcal{B} \leftarrow \text{NOTInspect}(X, (b^*, e], \mathcal{M}, \mathcal{B}, \lambda, \xi)$ 
return  $\mathcal{B}$ 

```

Appendix D: Empirical comparison between different variants of ESAC

In the following we compare the empirical performance of different variants of the ESAC algorithm. In all versions, seeded intervals are generated using Algorithm 4 with parameters α and K specified. The variants and configurations considered are:

- ESAC A: Algorithm 3 *without interval trimming* and with $\alpha = 2$, $K = 4$;
- ESAC B: Algorithm 1 with $\alpha = 2$, $K = 4$;
- ESAC C: Algorithm 1 with $\alpha = 3/2$, $K = 4$;
- ESAC D: Algorithm 2 with $\alpha = 3/2$, $K = 4$;
- ESAC E: Algorithm 1 without Narrowest-Over-Threshold choice of changepoint location and $\alpha = 3/2$, $K = 4$;
- ESAC F: Algorithm 1 with mid-point estimation and $\alpha = 3/2$, $K = 4$.

ESAC A is a mix of Algorithm 3 and Algorithm 1, in the sense that it tests for a changepoint at the midpoint of each seeded interval, but does not trim away intervals once a changepoint is detected. With ESAC E, a changepoint location is estimated by considering all seeded intervals in which a changepoint is detected, and not only the narrowest seeded intervals. This is achieved by replacing \mathcal{O}_{t^*} by $\mathcal{O}_{(s,e]}$ in Algorithm 1. In ESAC F, the estimated changepoint location v^* is replaced by $v^* = \lfloor (e_{m^*} + s_{m^*})/2 \rfloor$.

We have run a simulation with the exact same configuration as in Section 4.2. For changepoint detection, we have chosen the empirical penalty function $\tilde{\lambda}(t)$ as in Appendix B separately for each variant of ESAC. For changepoint estimation we have used the analytical penalty function $\tilde{\lambda}(t)$ as given in Appendix B. For each variant of ESAC and each configuration of parameters and changepoint regimes, Table 6 displays the average Hausdorff distance, average absolute estimation error of J and average run time in milliseconds. For each configuration of parameters and changepoint regimes, the minimum (and best) value of each of the performance measures is indicated in boldface.

Comparing ESAC A and B, one observes that testing only for a changepoint at the midpoint of a seeded interval results in a substantial improvement of run time but with a cost to statistical accuracy. The run time of ESAC B is roughly three to four times that of ESAC A, while the average Hausdorff distance and absolute estimation error of K of ESAC B are generally significantly larger than those of ESAC A, independently of the model configuration. This is likely due to ESAC A having lower power in detecting changepoints than ESAC B, as is indicated by ESAC A having higher estimation error of K . Comparing ESAC B and C, one observes a similar effect of decreasing α from 2 to 3/2. ESAC C has a run time almost twice that of ESAC B, while the average Hausdorff distance over all simulation setups is around half that of ESAC B. Comparing ESAC C and D, one observes that interval trimming substantially reduces statistical performance, with virtually no gain in terms of computational cost. Importantly, the estimation error of K is markedly higher for ESAC D, which indicates that interval trimming reduces power in detecting changepoints.

TABLE 6. Multiple changepoint estimation with different variants of ESAC.

Parameters				Hausdorff distance						$ \hat{J} - J $						Time in milliseconds						
n	p	Sparsity	K	A	B	C	D	E	F	A	B	C	D	E	F	A	B	C	D	E	F	
200	100	–	0	–	–	–	–	–	–	0.000	0.000	0.000	0.000	0.000	0.000	2.494	8.122	13.202	13.291	13.429	13.254	
200	100	Dense	2	29.557	7.248	7.197	19.046	7.046	12.934	0.453	0.104	0.088	0.414	0.093	0.110	2.720	8.088	12.539	12.329	13.826	12.549	
200	100	Sparse	2	5.172	1.615	1.658	6.885	1.525	5.886	0.085	0.020	0.012	0.150	0.019	0.025	2.822	7.838	12.347	12.036	13.794	12.580	
200	100	Mixed	2	17.593	5.016	5.006	13.105	5.087	9.727	0.274	0.065	0.054	0.280	0.067	0.068	2.615	7.857	12.385	12.130	13.766	12.474	
200	100	Dense	5	24.954	10.901	6.476	20.446	6.747	10.947	1.107	0.345	0.194	1.246	0.225	0.254	2.695	7.133	11.089	10.747	14.609	11.043	
200	100	Sparse	5	7.042	3.584	1.585	9.954	1.748	5.156	0.236	0.088	0.028	0.483	0.058	0.052	2.699	7.056	10.710	10.719	14.382	10.906	
200	100	Mixed	5	17.453	8.030	4.609	16.320	4.565	8.727	0.734	0.231	0.128	0.882	0.137	0.157	2.774	7.083	10.964	10.580	14.536	10.942	
200	1000	–	0	–	–	–	–	–	–	0.002	0.001	0.000	0.000	0.000	0.000	21.873	80.721	132.741	132.657	132.787	132.845	
200	1000	Dense	2	8.523	2.200	1.731	11.782	1.482	6.582	0.144	0.029	0.016	0.285	0.021	0.025	23.498	75.455	121.002	118.612	134.114	121.129	
200	1000	Sparse	2	1.772	1.166	0.972	4.981	0.751	4.916	0.024	0.011	0.004	0.121	0.009	0.008	23.417	73.991	119.314	116.363	133.431	118.794	
200	1000	Mixed	2	4.570	2.151	1.779	9.092	1.564	6.183	0.081	0.024	0.015	0.210	0.019	0.021	23.233	74.204	119.315	116.722	133.449	119.462	
200	1000	Dense	5	11.205	3.319	2.201	16.493	1.929	6.409	0.412	0.090	0.043	0.919	0.059	0.053	23.379	65.516	103.870	100.549	139.538	103.803	
200	1000	Sparse	5	3.561	1.982	0.933	9.515	0.736	4.507	0.092	0.045	0.011	0.429	0.025	0.021	23.647	65.101	101.906	98.008	136.001	101.782	
200	1000	Mixed	5	8.422	3.099	1.754	14.095	1.373	5.419	0.274	0.072	0.033	0.682	0.040	0.052	23.416	65.203	102.504	99.180	138.009	102.491	
200	5000	–	0	–	–	–	–	–	–	0.000	0.002	0.003	0.003	0.003	0.003	110.148	425.285	700.419	700.541	698.719	702.664	
200	5000	Dense	2	6.299	1.293	1.150	5.607	0.747	5.082	0.107	0.010	0.005	0.130	0.009	0.007	118.414	385.540	619.304	599.710	702.220	619.071	
200	5000	Sparse	2	1.586	0.909	0.812	2.787	0.406	4.261	0.022	0.005	0.001	0.049	0.003	0.009	117.881	380.432	607.773	591.900	702.059	608.943	
200	5000	Mixed	2	5.353	1.105	1.081	5.124	0.668	5.067	0.085	0.007	0.005	0.115	0.007	0.016	119.099	382.842	615.231	595.623	698.241	613.451	
200	5000	Dense	5	8.858	2.516	0.836	9.285	0.544	4.386	0.290	0.054	0.006	0.448	0.016	0.021	118.777	334.106	516.330	498.218	712.661	517.062	
200	5000	Sparse	5	3.073	1.200	0.680	4.043	0.341	3.737	0.080	0.019	0.001	0.160	0.007	0.015	118.780	328.529	507.473	486.259	703.117	507.189	
200	5000	Mixed	5	6.073	1.582	0.706	7.221	0.453	4.124	0.190	0.032	0.004	0.321	0.014	0.019	118.770	330.622	511.869	489.977	705.463	511.883	
500	100	–	0	–	–	–	–	–	–	0.001	0.000	0.001	0.001	0.001	0.001	6.205	24.287	40.742	40.820	40.817	40.840	
500	100	Dense	2	67.900	29.573	10.152	42.356	10.293	22.459	0.384	0.148	0.047	0.379	0.062	0.053	6.580	23.749	38.286	37.800	41.399	38.353	
500	100	Sparse	2	21.915	8.268	2.361	9.141	2.005	11.876	0.119	0.035	0.001	0.075	0.007	0.008	6.589	23.215	37.525	36.811	41.112	37.503	
500	100	Mixed	2	49.745	17.831	5.829	27.407	6.041	17.855	0.277	0.087	0.021	0.240	0.032	0.036	6.526	23.268	37.893	37.271	41.240	37.939	
500	100	Dense	5	48.052	21.964	11.901	50.218	12.038	22.616	0.820	0.272	0.125	1.265	0.148	0.176	6.665	21.533	34.640	33.852	43.034	34.689	
500	100	Sparse	5	18.105	7.235	2.312	17.084	2.506	10.364	0.266	0.071	0.007	0.334	0.046	0.032	6.652	21.166	33.671	32.940	42.543	33.844	
500	100	Mixed	5	34.329	14.720	6.449	33.383	6.254	15.993	0.567	0.174	0.056	0.755	0.086	0.086	6.708	21.385	34.159	33.330	43.037	34.114	
500	1000	–	0	–	–	–	–	–	–	0.000	0.003	0.000	0.000	0.000	0.000	58.259	241.550	408.466	410.328	408.340	411.164	
500	1000	Dense	2	35.257	6.585	4.104	24.037	3.594	14.977	0.203	0.026	0.014	0.254	0.018	0.021	62.356	231.143	378.515	371.903	410.607	379.165	
500	1000	Sparse	2	6.619	2.473	1.603	7.812	1.050	10.553	0.034	0.004	0.000	0.078	0.006	0.001	62.345	227.051	372.265	363.301	411.828	371.272	
500	1000	Mixed	2	21.890	6.052	2.034	15.490	1.473	12.472	0.117	0.023	0.002	0.150	0.008	0.006	62.723	228.743	372.856	366.669	410.545	374.385	
500	1000	Dense	5	28.396	7.049	2.960	39.687	2.806	12.167	0.438	0.066	0.017	0.912	0.040	0.025	63.259	207.844	336.695	327.587	427.493	335.335	
500	1000	Sparse	5	8.535	3.037	1.769	14.935	1.211	9.094	0.096	0.020	0.003	0.278	0.025	0.011	63.527	204.245	328.075	317.877	416.863	326.975	
500	1000	Mixed	5	17.612	4.813	3.147	26.652	2.371	11.192	0.274	0.034	0.017	0.572	0.033	0.025	63.396	205.635	333.017	323.492	423.150	332.866	
500	5000	–	0	–	–	–	–	–	–	0.000	0.000	0.000	0.000	0.000	0.000	316.528	1328.522	2242.270	2245.298	2249.252	2236.030	
500	5000	Dense	2	16.733	5.984	2.294	13.965	1.390	11.295	0.088	0.023	0.002	0.132	0.007	0.003	336.784	1262.177	2023.982	1994.241	2267.927	2031.555	
500	5000	Sparse	2	4.798	1.625	1.951	3.990	0.859	10.402	0.024	0.001	0.002	0.022	0.003	0.008	334.060	1241.024	2009.202	1957.339	2269.248	1999.724	
500	5000	Mixed	2	14.071	5.259	1.946	8.262	1.092	11.044	0.068	0.018	0.002	0.068	0.009	0.009	331.958	1250.821	2026.691	1978.321	2270.342	2019.991	
500	5000	Dense	5	13.681	5.718	2.015	20.418	1.283	10.137	0.182	0.054	0.003	0.408	0.029	0.018	335.042	1118.323	1771.210	1701.822	2302.847	1761.599	
500	5000	Sparse	5	3.990	2.460	1.726	6.917	0.877	8.494	0.038	0.012	0.006	0.098	0.019	0.020	334.404	1085.380	1720.665	1656.142	2260.874	1714.863	
500	5000	Mixed	5	9.622	4.317	1.925	14.145	0.908	9.686	0.120	0.036	0.001	0.255	0.021	0.021	331.365	1095.025	1731.004	1666.874	2258.874	1733.001	
Average				16.453	5.941	2.990	15.602	2.660	9.631	0.210	0.056	0.023	0.324	0.034	0.036							

Comparing ESAC C and E, one observes that using a Narrowest-over-Threshold method to estimate changepoints (as opposed to considering seeded intervals of all widths) has a mixed effect on statistical performance and a positive effect on the computational cost. In terms of Hausdorff distance, ESAC E tends to slightly outperform ESAC C, while the converse is true when considering estimation error of K . In terms of run time, ESAC C slightly outperforms ESAC E, especially when there are many changepoints. Lastly, comparing ESAC C and F, one observes that estimating changepoints using the penalized score improves estimation accuracy compared to estimating changepoints by taking a mid-point of a seeded interval. For ESAC C, The average Hausdorff distance over all simulations is around one third that of ESAC F. Somewhat less pronounced is the difference in estimation error of K , where ESAC C also outperforms ESAC F.

Appendix E: Some more simulations

E.1. Extended table from Section 4.2

Tables 7 and 8 display the results from running the simulation study conducted in Section 4.2 also for $p = 5000$, so that p ranges over $\{100, 1000, 5000\}$.

E.2. Simulations with randomly drawn changes in mean

Table 9 and Tables 10, 11 respectively display the results of re-running the simulations in Sections 4.1 and 4.2 with the modification that the changes in the mean-vector are drawn randomly. More specifically, for each changepoint we have taken the change in mean θ to satisfy $\theta_{1:k} \propto (Z^\top, 0_{p-k}^\top)$ where $Z \sim N_k(0, 1)$. Apart from this modification, the simulation setups are identical to the ones in Sections 4.1 and 4.2, including the magnitudes of the changes in mean.

In the single changepoint case, ESAC displays a slightly larger variability in performance compared to the simulation where changes in mean are evenly spread across the affected coordinates. Averaging over all values of n , p and k , one observes that the MSE of ESAC has increased slightly in comparison with Table 1. Meanwhile, the opposite is true for the competing methods. Still, Table 9 shows that the performance of ESAC is competitive also when the change in the mean vector is not evenly spread across the affected coordinates.

The same conclusions hold in the multiple changepoint case. Tables 10 and 11 show that the performance of ESAC is slightly worse than in Table 3. Still, the performance of ESAC is competitive.

E.3. Single changepoint detection

Here we investigate the power of each method when testing for the presence of a single changepoint. It is assumed known that there is at most one changepoint in

TABLE 7
Multiple changepoints, Hausdorff distance.

Parameters				Hausdorff distance						
n	p	Sparsity	J	ESAC	Pilliat	Inspect	SBS	SUBSET	DC	Kaul et al
100	100	–	0	–	–	–	–	–	–	–
100	100	Dense	2	0.76	9.40	2.19	41.77	1.08	45.30	2.19
100	100	Sparse	2	0.63	4.90	1.23	41.07	0.61	15.77	1.16
100	100	Mixed	2	0.53	7.75	2.11	41.01	1.23	32.48	2.26
100	100	Dense	5	0.48	10.53	2.87	45.06	1.35	34.02	2.72
100	100	Sparse	5	0.37	5.26	1.64	46.44	1.06	19.37	1.57
100	100	Mixed	5	0.43	7.75	2.61	45.64	1.19	25.39	2.51
100	1000	–	0	–	–	–	–	–	–	–
100	1000	Dense	2	0.36	5.20	1.58	35.46	0.42	–	9.18
100	1000	Sparse	2	0.30	3.88	3.02	39.76	0.38	–	3.07
100	1000	Mixed	2	0.48	4.29	2.38	40.65	0.58	–	6.20
100	1000	Dense	5	0.25	6.06	2.08	42.13	0.64	–	2.19
100	1000	Sparse	5	0.23	3.85	3.09	44.72	0.65	–	3.07
100	1000	Mixed	5	0.22	4.41	2.86	46.31	0.78	–	2.98
100	5000	–	0	–	–	–	–	–	–	–
100	5000	Dense	2	0.46	4.20	1.99	36.29	0.57	–	18.24
100	5000	Sparse	2	0.49	3.16	6.16	39.51	0.28	–	6.34
100	5000	Mixed	2	0.40	3.82	3.48	37.96	0.30	–	9.04
100	5000	Dense	5	0.16	5.04	2.77	41.28	0.75	–	3.80
100	5000	Sparse	5	0.33	2.97	5.12	42.84	0.58	–	5.12
100	5000	Mixed	5	0.17	3.88	3.87	42.86	0.66	–	4.56
200	100	–	0	–	–	–	–	–	–	–
200	100	Dense	2	1.25	16.44	3.10	58.53	2.10	65.00	4.89
200	100	Sparse	2	0.97	7.07	1.87	47.77	1.26	12.07	1.72
200	100	Mixed	2	1.09	10.94	2.76	51.05	1.45	45.49	5.03
200	100	Dense	5	1.14	16.99	4.44	57.71	2.26	51.17	4.07
200	100	Sparse	5	0.89	7.57	2.19	56.82	2.02	21.56	2.02
200	100	Mixed	5	0.74	13.21	2.74	60.62	2.16	39.22	2.42
200	1000	–	0	–	–	–	–	–	–	–
200	1000	Dense	2	1.13	7.93	2.51	49.75	0.90	–	34.02
200	1000	Sparse	2	0.88	4.19	5.56	49.73	0.60	–	7.70
200	1000	Mixed	2	0.94	6.62	4.05	50.51	0.98	–	16.65
200	1000	Dense	5	0.61	9.81	3.51	54.88	1.54	–	4.57
200	1000	Sparse	5	0.44	5.12	6.22	55.74	1.52	–	6.15
200	1000	Mixed	5	0.50	7.50	4.59	57.64	1.47	–	5.13
200	5000	–	0	–	–	–	–	–	–	–
200	5000	Dense	2	0.83	7.64	3.62	51.81	0.60	–	52.87
200	5000	Sparse	2	0.66	3.63	10.41	58.17	0.43	–	12.53
200	5000	Mixed	2	0.76	5.93	6.30	54.42	0.46	–	25.45
200	5000	Dense	5	0.51	9.64	3.73	51.95	1.17	–	9.91
200	5000	Sparse	5	0.58	4.67	9.60	54.29	1.41	–	9.65
200	5000	Mixed	5	0.53	7.06	6.34	52.05	1.01	–	8.43
Average				0.60	6.90	3.74	47.89	1.01	33.90	7.13

the simulated data, and thus no multiple changepoint search method like Binary Segmentation or Seeded Binary Segmentation is used for any of the methods. Instead, we have for each method computed the corresponding test statistic for a single changepoint on the whole generated data set X , using e.g. $S_{\gamma, (0, n]}^v$ in (3) for ESAC. Our simulations are run with the same setup as in Section 4.1, with the exception of a slightly lower signal strength to avoid 0% testing error. We adjust φ such that $\Delta\varphi^2 = n \|\theta\|_2^2 / 5 = 1.8^2 r(k)$ for each combination of n, p and

TABLE 8
Multiple changepoints, estimation of J .

Parameters				$ \hat{J} - J $						
n	p	Sparsity	J	ESAC	Pilliat	Inspect	SBS	SUBSET	DC	Kaul et al
100	100	–	0	0.01	0.00	0.01	0.02	0.04	0.01	0.01
100	100	Dense	2	0.01	0.40	0.04	1.19	0.08	1.30	0.04
100	100	Sparse	2	0.01	0.19	0.03	1.09	0.04	0.46	0.03
100	100	Mixed	2	0.00	0.31	0.05	1.14	0.06	0.94	0.05
100	100	Dense	5	0.01	1.02	0.14	3.77	0.19	3.24	0.14
100	100	Sparse	5	0.01	0.43	0.11	3.76	0.21	2.13	0.11
100	100	Mixed	5	0.01	0.72	0.14	3.76	0.22	2.67	0.14
100	1000	–	0	0.00	0.00	0.01	0.27	0.05	–	0.01
100	1000	Dense	2	0.00	0.24	0.03	0.98	0.03	–	0.03
100	1000	Sparse	2	0.00	0.16	0.10	1.05	0.04	–	0.10
100	1000	Mixed	2	0.01	0.18	0.06	1.05	0.10	–	0.06
100	1000	Dense	5	0.00	0.56	0.10	3.62	0.17	–	0.10
100	1000	Sparse	5	0.00	0.32	0.24	3.71	0.16	–	0.24
100	1000	Mixed	5	0.00	0.38	0.19	3.77	0.17	–	0.19
100	5000	–	0	0.01	0.01	0.02	1.36	0.09	–	0.02
100	5000	Dense	2	0.01	0.17	0.04	0.78	0.09	–	0.04
100	5000	Sparse	2	0.01	0.14	0.18	0.85	0.03	–	0.18
100	5000	Mixed	2	0.01	0.15	0.10	0.80	0.03	–	0.10
100	5000	Dense	5	0.00	0.51	0.17	3.53	0.17	–	0.17
100	5000	Sparse	5	0.01	0.25	0.41	3.61	0.15	–	0.41
100	5000	Mixed	5	0.00	0.35	0.29	3.62	0.17	–	0.29
200	100	–	0	0.01	0.01	0.01	0.04	0.06	0.01	0.01
200	100	Dense	2	0.01	0.38	0.03	0.89	0.07	0.98	0.03
200	100	Sparse	2	0.00	0.17	0.02	0.67	0.04	0.19	0.02
200	100	Mixed	2	0.00	0.25	0.02	0.75	0.03	0.69	0.02
200	100	Dense	5	0.01	1.01	0.09	3.02	0.19	2.64	0.09
200	100	Sparse	5	0.01	0.35	0.06	2.77	0.22	1.28	0.06
200	100	Mixed	5	0.00	0.69	0.07	2.93	0.20	2.09	0.07
200	1000	–	0	0.00	0.00	0.01	0.31	0.05	–	0.01
200	1000	Dense	2	0.01	0.20	0.02	0.64	0.03	–	0.02
200	1000	Sparse	2	0.00	0.09	0.10	0.64	0.04	–	0.10
200	1000	Mixed	2	0.00	0.17	0.05	0.65	0.03	–	0.05
200	1000	Dense	5	0.00	0.57	0.06	2.83	0.18	–	0.06
200	1000	Sparse	5	0.00	0.25	0.24	2.74	0.18	–	0.24
200	1000	Mixed	5	0.00	0.39	0.15	2.82	0.21	–	0.15
200	5000	–	0	0.01	0.00	0.03	2.02	0.02	–	0.03
200	5000	Dense	2	0.01	0.21	0.03	0.51	0.03	–	0.03
200	5000	Sparse	2	0.00	0.09	0.16	0.56	0.05	–	0.16
200	5000	Mixed	2	0.00	0.15	0.08	0.53	0.02	–	0.08
200	5000	Dense	5	0.00	0.54	0.08	2.42	0.18	–	0.08
200	5000	Sparse	5	0.01	0.22	0.39	2.57	0.19	–	0.39
200	5000	Mixed	5	0.00	0.37	0.23	2.50	0.15	–	0.23
Average				0.01	0.30	0.10	1.82	0.11	1.33	0.10

k . Note that the method of [9] is not included, as this method only serves as a “post-detection” estimator.

Similar to the version of ESAC given in Algorithm 3, the Pilliat method only tests for a changepoint in the midpoint of any seeded interval (s, e) . This time saving trick does not affect the theoretical guarantees of neither ESAC nor Pilliat in the multiple changepoint situation because intervals for both methods

TABLE 9
Single changepoint estimation MSE.

Parameters					Mean Squared Error					
n	p	k	η	φ	ESAC	Inspect	SBS	SUBSET	DC	Kaul et al.
200	100	1	40	1.40	10.1	25.7	70.4	39.2	9.4	48.4
200	100	5	40	2.00	4.6	2.4	40.0	2.6	6.4	4.1
200	100	24	40	1.90	46.6	20.3	977.3	125.1	144.4	157.0
200	100	100	40	1.90	53.0	147.2	1507.2	307.4	1561.1	244.3
200	1000	1	40	1.52	5.1	95.2	34.9	27.5	5.2	13.1
200	1000	10	40	2.93	1.4	0.6	2.5	0.5	3.3	0.5
200	1000	73	40	3.37	6.1	2.0	510.3	4.9	14.3	0.5
200	1000	1000	40	3.37	3.7	544.8	1546.1	3.7	195.2	11200.0
200	5000	1	40	1.60	38.9	481.5	19.3	47.6	143.7	8.9
200	5000	18	40	4.00	1.1	0.5	0.9	0.6	1.3	0.1
200	5000	163	40	5.04	3.4	0.7	286.1	3.4	17.6	0.1
200	5000	5000	40	5.04	3.8	1299.5	1548.5	3.8	5.8	16390.4
500	100	1	100	0.92	48.8	86.9	130.1	242.1	48.0	49.3
500	100	5	100	1.31	14.2	13.3	49.8	12.5	39.0	11.7
500	100	25	100	1.25	223.3	63.7	5044.6	746.3	662.8	412.3
500	100	100	100	1.25	446.9	727.1	9526.7	1953.2	8495.3	1579.3
500	1000	1	100	1.00	30.1	200.6	30.0	125.3	29.3	49.6
500	1000	10	100	1.90	4.7	4.3	11.4	2.7	14.7	3.9
500	1000	79	100	2.22	14.6	10.5	1854.2	13.3	132.7	2.8
500	1000	1000	100	2.22	22.4	2083.8	9755.1	63.5	2258.4	99661.6
500	5000	1	100	1.05	26.4	1290.1	36.0	75.3	266.4	39.2
500	5000	18	100	2.58	2.2	1.4	6.9	1.2	9.1	1.4
500	5000	177	100	3.32	13.9	1.9	812.9	13.9	109.1	0.5
500	5000	5000	100	3.32	14.9	6546.6	9921.2	14.9	24.7	136207.7
Average MSE					43.3	568.8	1821.8	159.6	591.6	11087.0

are generated such that any changepoint will be close to a midpoint of some interval (s, e) . In this section, however, we are concerned with testing for a changepoint over a single interval (i.e. $(s, e) = (0, n)$), in which case testing only for a changepoint in the midpoint can lead to great efficiency losses whenever the true changepoint is far from the midpoint. To obtain fair and meaningful power comparisons with the remaining methods in the simulation study, we have modified the test statistic from the Pilliat method to test for a changepoint in all time points $v = 1, \dots, n - 1$.

For any testing procedure, there is a trade-off between Type I and Type II errors. In order to have precise control over the Type I error of each method, we have run the competing methods with empirically chosen penalty parameters. Each method is calibrated to have Type I error at most 1% based on $N = 1000$ Monte Carlo simulations. The methods ESAC and Pilliat, unlike the remaining methods, combine several test statistics to test for a changepoint, resulting in a multiple testing situation. For ESAC we have adjusted for the multiple testing by using the empirical penalty function $\tilde{\gamma}$ as defined in Appendix B. Similarly, we have for the Pilliat method chosen thresholds for two of its three constituent tests by Monte Carlo simulating the leading constant in the theoretical thresholds and applied a Bonferroni correction. For the last test statistic used in the Pilliat method (the Berk Jones statistic), we have used the theoretical threshold provided in the paper. For Inspect, we have chosen the detection

TABLE 10
Multiple changepoints, Hausdorff distance.

Parameters				Hausdorff distance						
n	p	Sparsity	J	ESAC	Pilliat	Inspect	SBS	SUBSET	DC	Kaul et al
100	100	–	0	–	–	–	–	–	–	–
100	100	Dense	2	0.81	10.45	1.51	38.41	1.35	29.30	1.45
100	100	Sparse	2	0.47	4.21	1.42	42.27	0.64	13.94	1.41
100	100	Mixed	2	0.49	7.41	1.70	41.37	0.96	23.77	1.72
100	100	Dense	5	0.59	10.46	2.58	45.82	1.44	26.70	2.50
100	100	Sparse	5	0.38	4.34	1.73	45.50	1.03	17.66	1.72
100	100	Mixed	5	0.42	7.12	2.02	46.36	1.30	22.44	1.98
100	1000	–	0	–	–	–	–	–	–	–
100	1000	Dense	2	0.39	5.37	0.95	38.13	0.30	–	3.26
100	1000	Sparse	2	0.20	2.05	3.56	39.28	0.29	–	3.59
100	1000	Mixed	2	0.33	3.58	2.68	37.90	0.52	–	4.09
100	1000	Dense	5	0.20	5.54	1.35	42.06	0.92	–	1.37
100	1000	Sparse	5	0.29	2.00	3.80	45.90	0.76	–	3.81
100	1000	Mixed	5	0.18	3.74	2.84	44.10	0.77	–	2.87
100	5000	–	0	–	–	–	–	–	–	–
100	5000	Dense	2	0.41	4.51	1.40	37.89	0.61	–	10.60
100	5000	Sparse	2	0.58	1.12	6.19	38.06	0.26	–	6.11
100	5000	Mixed	2	0.57	2.81	4.53	37.60	0.34	–	7.52
100	5000	Dense	5	0.19	4.47	1.92	39.72	0.62	–	2.47
100	5000	Sparse	5	0.30	1.04	6.07	42.51	0.74	–	6.07
100	5000	Mixed	5	0.30	2.75	3.95	42.89	0.74	–	4.28
200	100	–	0	–	–	–	–	–	–	–
200	100	Dense	2	1.25	17.11	2.32	48.00	1.97	30.40	2.56
200	100	Sparse	2	0.86	4.74	2.06	45.19	1.22	10.58	1.95
200	100	Mixed	2	1.22	11.59	2.24	49.75	1.52	21.51	3.06
200	100	Dense	5	0.89	17.18	2.71	53.54	2.27	31.61	2.43
200	100	Sparse	5	0.62	5.32	2.35	58.81	2.02	20.02	2.30
200	100	Mixed	5	0.78	11.14	2.60	56.26	1.93	28.61	2.39
200	1000	–	0	–	–	–	–	–	–	–
200	1000	Dense	2	0.88	10.04	1.48	45.40	1.31	–	12.69
200	1000	Sparse	2	0.60	2.10	7.67	41.98	0.74	–	7.78
200	1000	Mixed	2	0.60	4.88	3.80	48.10	0.66	–	9.56
200	1000	Dense	5	0.70	10.09	1.90	52.75	1.63	–	2.00
200	1000	Sparse	5	0.40	2.15	6.45	54.24	1.61	–	6.42
200	1000	Mixed	5	0.52	6.51	4.19	53.14	1.46	–	4.45
200	5000	–	0	–	–	–	–	–	–	–
200	5000	Dense	2	0.70	7.01	3.09	55.55	0.41	–	30.90
200	5000	Sparse	2	0.94	1.69	12.00	55.36	0.78	–	12.23
200	5000	Mixed	2	0.57	4.15	7.58	55.47	0.52	–	19.30
200	5000	Dense	5	0.61	9.67	2.85	48.13	1.34	–	5.65
200	5000	Sparse	5	0.34	1.40	12.25	54.68	1.26	–	12.19
200	5000	Mixed	5	0.57	5.27	7.14	51.84	1.16	–	8.67
Average				0.56	5.97	3.75	46.50	1.04	23.04	5.08

threshold ξ to be the 10th largest sparse projection $\max_{0 < b < n} (\hat{v}_\lambda^{(0,n)})^\top T_b^{(0,n)}$, where $\lambda = \{\log(p \log n) / 2\}^{1/2}$ (see Appendix C), over $N = 1000$ data sets with no changepoints. For SUBSET we have used the function for choosing the penalty parameter β provided by the author, with the remaining penalty parameters at their recommended values, also using the 10th largest value out of $N = 1000$ Monte Carlo samples. For Sparsified Binary Segmentation we have chosen the

TABLE 11
Multiple changepoints, estimation of J .

Parameters				$ \hat{J} - J $						
n	p	Sparsity	J	ESAC	Pilliat	Inspect	SBS	SUBSET	DC	Kaul et al
100	100	–	0	0.01	0.00	0.01	0.02	0.04	0.01	0.01
100	100	Dense	2	0.01	0.43	0.04	1.08	0.08	0.87	0.04
100	100	Sparse	2	0.01	0.16	0.04	1.12	0.05	0.37	0.04
100	100	Mixed	2	0.00	0.27	0.04	1.12	0.05	0.68	0.04
100	100	Dense	5	0.01	1.07	0.14	3.76	0.21	2.72	0.14
100	100	Sparse	5	0.01	0.35	0.12	3.73	0.20	1.97	0.12
100	100	Mixed	5	0.01	0.65	0.12	3.74	0.21	2.38	0.12
100	1000	–	0	0.01	0.00	0.01	0.28	0.03	–	0.01
100	1000	Dense	2	0.00	0.24	0.02	1.01	0.02	–	0.02
100	1000	Sparse	2	0.00	0.07	0.11	1.05	0.05	–	0.11
100	1000	Mixed	2	0.00	0.13	0.07	0.99	0.03	–	0.07
100	1000	Dense	5	0.00	0.55	0.05	3.62	0.19	–	0.05
100	1000	Sparse	5	0.01	0.12	0.27	3.73	0.17	–	0.27
100	1000	Mixed	5	0.00	0.30	0.19	3.66	0.18	–	0.19
100	5000	–	0	0.01	0.01	0.03	1.32	0.11	–	0.03
100	5000	Dense	2	0.01	0.18	0.03	0.79	0.13	–	0.03
100	5000	Sparse	2	0.02	0.03	0.19	0.79	0.05	–	0.19
100	5000	Mixed	2	0.02	0.10	0.13	0.78	0.03	–	0.13
100	5000	Dense	5	0.00	0.47	0.11	3.52	0.16	–	0.11
100	5000	Sparse	5	0.01	0.06	0.50	3.60	0.18	–	0.50
100	5000	Mixed	5	0.01	0.22	0.30	3.60	0.17	–	0.30
200	100	–	0	0.01	0.00	0.01	0.04	0.11	0.02	0.01
200	100	Dense	2	0.00	0.38	0.02	0.69	0.09	0.46	0.02
200	100	Sparse	2	0.01	0.10	0.03	0.62	0.06	0.16	0.03
200	100	Mixed	2	0.01	0.24	0.03	0.68	0.06	0.30	0.03
200	100	Dense	5	0.00	0.97	0.05	2.76	0.24	1.77	0.05
200	100	Sparse	5	0.01	0.22	0.08	2.81	0.25	1.20	0.08
200	100	Mixed	5	0.01	0.54	0.06	2.78	0.26	1.59	0.06
200	1000	–	0	0.00	0.00	0.01	0.34	0.07	–	0.01
200	1000	Dense	2	0.01	0.25	0.01	0.57	0.05	–	0.01
200	1000	Sparse	2	0.01	0.03	0.12	0.60	0.09	–	0.12
200	1000	Mixed	2	0.00	0.11	0.05	0.62	0.03	–	0.05
200	1000	Dense	5	0.00	0.59	0.03	2.69	0.20	–	0.03
200	1000	Sparse	5	0.00	0.06	0.25	2.72	0.19	–	0.25
200	1000	Mixed	5	0.00	0.28	0.14	2.71	0.18	–	0.14
200	5000	–	0	0.01	0.01	0.03	1.98	0.01	–	0.03
200	5000	Dense	2	0.00	0.18	0.02	0.53	0.03	–	0.02
200	5000	Sparse	2	0.01	0.02	0.20	0.56	0.04	–	0.20
200	5000	Mixed	2	0.00	0.09	0.11	0.52	0.04	–	0.11
200	5000	Dense	5	0.00	0.56	0.06	2.39	0.17	–	0.06
200	5000	Sparse	5	0.01	0.02	0.49	2.55	0.18	–	0.49
200	5000	Mixed	5	0.01	0.23	0.26	2.48	0.17	–	0.26
Average				0.01	0.25	0.11	1.78	0.12	1.04	0.11

threshold π_T in the same way as in Section 4.1, also using the 10th largest among $N = 1000$ Monte Carlo samples. For the Double CUSUM algorithm we have used the input parameter $\varphi = -1$ and chosen the threshold value to be the 10th largest double CUSUM statistic over $N = 1000$ Monte Carlo simulated data sets without any changepoints.

For each method considered and each configuration of parameters, Table 12 displays the average detection rate and average running time in milliseconds. For each configuration of parameters, the best value of the detection rate and the run time is indicated in boldface (when there are no changepoints, boldface indicates the detection rate closest to 1% from below). In terms of statistical power, Table 12 demonstrates that ESAC, Pilliat and SUBSET are the only methods with competitive power across all sparsity regimes and combinations of n and p . Pilliat has the highest power in seven out of the 24 different combinations of parameters with a changepoint, while the same number is three for ESAC and one for SUBSET. Averaging over the 24 combinations of parameters, Pilliat and ESAC have the highest over-all power. The Pilliat method has a slight edge over ESAC, and SUBSET in third place. In comparison, Inspect has high detection power only for $k = \lceil p^{1/3} \rceil$, and with performance seemingly deteriorating when p grows. Double CUSUM has excellent power for detecting dense changepoints, but fails to detect sparse changepoints, especially when $k = 1$ or p is large. Sparsified Binary Segmentation has high power for sparse changepoints (especially when $k = 1$), but fails completely to detect dense changepoints. In terms of run time, ESAC is again the clear winner, with Pilliat as the runner-up. We remark again that SUBSET is the only method not implemented in C or C++, giving the other methods an advantage when comparing run times. We also remark that the run time of the noise level scaling by MAD estimates is not included in the run times of ESAC, Inspect, Pilliat and SUBSET, as the run time of the scaling dominates the run time of ESAC, SUBSET and Pilliat. The run time of the MAD scaling is however included in the run times of the Double CUSUM and Sparsified Binary Segmentation algorithms, as the implementations of these algorithms do not offer an option to disable the MAD scaling.

It is interesting to note that the power of ESAC, Pilliat and SUBSET seems to grow with n . This might be due to the SNR of the simulated changepoints being proportional to the detection boundary for multiple changepoints, which grows faster with n than the minimax testing rate for a single changepoint, see Liu, Gao and Samworth [13].

Appendix F: Auxiliary lemmas

Lemma F.1. For any $a \geq 0$, define $\nu_a = \mathbb{E}(Z^2 \mid |Z| \geq a)$ where $Z \sim N(0, 1)$. Then

$$a^2 + 1 \leq \nu_a \leq a^2 + 2.$$

Proof. The second inequality follows from Lemma 4 in Liu, Gao and Samworth [13]. For the first inequality, let $\bar{\Phi}(x) = \int_x^\infty \varphi(t)dt$, where $\varphi(\cdot)$ denotes the density function of a standard normal distribution. If $a > 0$, we have that

$$\nu_a - 1 - a^2 = a \frac{\varphi(a)}{\bar{\Phi}(a)} - a^2$$

TABLE 12
Single changepoint detection.

Parameters					Detection rate						Time in milliseconds						
n	p	k	η	φ	ESAC	Pilliat	Inspect	SBS	SUBSET	DC	ESAC	Pilliat	Inspect	SBS	SUBSET	DC	
200	100	-	-	-	0.016	0.013	0.008	0.015	0.017	0.007	0.2	0.8	2.2	11.2	2.1	9.5	
200	100	1	$n/5$	1.12	0.849	0.815	0.232	0.892	0.886	0.082	0.1	0.3	2.1	9.8	1.7	8.6	
200	100	5	$n/5$	1.60	0.952	0.965	0.911	0.690	0.962	0.775	0.1	0.2	2.0	9.4	1.6	8.6	
200	100	24	$n/5$	1.52	0.696	0.749	0.666	0.075	0.567	0.813	0.1	0.4	2.0	12.6	1.6	8.5	
200	100	100	$n/5$	1.52	0.675	0.740	0.550	0.038	0.543	0.819	0.1	0.4	2.1	9.4	1.7	8.8	
200	1000	-	-	-	0.008	0.009	0.010	0.006	0.010	0.008	1.8	10.6	42.8	88.2	12.5	87.5	
200	1000	1	$n/5$	1.22	0.809	0.796	0.003	0.853	0.841	0.018	1.1	4.0	42.5	88.4	13.8	89.0	
200	1000	10	$n/5$	2.35	0.994	0.995	0.498	0.649	0.987	0.299	0.9	2.2	42.4	88.1	12.7	90.3	
200	1000	73	$n/5$	2.70	0.823	0.880	0.645	0.046	0.788	0.900	1.1	3.3	42.4	88.0	13.0	89.6	
200	1000	1000	$n/5$	2.70	0.805	0.868	0.390	0.014	0.783	0.918	1.1	3.4	42.5	87.7	13.3	89.1	
200	5000	-	-	-	0.007	0.008	0.003	0.012	0.012	0.012	11.6	65.4	216.7	449.8	78.0	458.9	
200	5000	1	$n/5$	1.28	0.782	0.775	0.000	0.856	0.797	0.018	7.1	26.0	215.9	447.3	77.7	455.9	
200	5000	18	$n/5$	3.20	0.996	1.000	0.011	0.698	0.997	0.182	5.6	17.4	215.0	446.0	78.0	454.6	
200	5000	163	$n/5$	4.03	0.911	0.925	0.249	0.038	0.898	0.911	6.7	18.5	214.9	441.5	79.1	457.1	
200	5000	5000	$n/5$	4.03	0.897	0.893	0.119	0.013	0.868	0.934	8.8	20.8	214.2	440.8	78.6	457.9	
500	100	-	-	-	0.007	0.005	0.007	0.015	0.015	0.008	0.5	2.1	5.6	16.0	4.2	16.0	
500	100	1	$n/5$	0.74	0.944	0.897	0.288	0.973	0.968	0.069	0.2	0.6	5.4	16.5	3.8	15.7	
500	100	5	$n/5$	1.04	0.978	0.988	0.973	0.882	0.987	0.821	0.2	0.5	5.4	16.7	3.6	15.5	
500	100	25	$n/5$	1.00	0.809	0.785	0.755	0.120	0.718	0.852	0.3	0.8	5.4	15.9	3.6	15.5	
500	100	100	$n/5$	1.00	0.796	0.776	0.627	0.062	0.699	0.853	0.3	0.8	5.4	16.0	3.7	15.8	
500	1000	-	-	-	0.004	0.004	0.005	0.014	0.010	0.008	4.4	26.4	261.9	147.0	32.8	163.2	
500	1000	1	$n/5$	0.80	0.924	0.876	0.000	0.963	0.942	0.026	2.5	8.8	259.7	147.4	32.4	163.4	
500	1000	10	$n/5$	1.52	0.994	0.998	0.551	0.860	0.994	0.357	2.3	5.7	259.0	148.1	32.4	164.7	
500	1000	79	$n/5$	1.78	0.936	0.926	0.711	0.073	0.868	0.925	2.8	7.6	260.5	148.6	32.5	165.2	
500	1000	1000	$n/5$	1.78	0.927	0.920	0.466	0.023	0.877	0.965	2.5	8.1	260.4	149.1	32.3	165.6	
500	5000	-	-	-	0.006	0.002	0.010	0.008	0.006	0.011	25.9	158.2	1343.5	769.4	182.0	955.7	
500	5000	1	$n/5$	0.84	0.939	0.890	0.000	0.958	0.947	0.017	13.7	46.6	1341.9	762.7	184.3	908.5	
500	5000	18	$n/5$	2.06	1.000	1.000	0.010	0.907	1.000	0.204	12.0	28.3	1341.4	763.1	185.4	908.1	
500	5000	177	$n/5$	2.66	0.964	0.959	0.455	0.045	0.951	0.956	12.8	37.7	1342.3	762.1	184.1	908.7	
500	5000	5000	$n/5$	2.66	0.953	0.956	0.243	0.009	0.956	0.982	13.0	37.5	1342.8	757.8	179.5	907.7	
Average detection rate					0.890	0.891	0.390	0.447	0.868	0.571							

$$= a \left\{ \frac{\varphi(a)}{\overline{\Phi}(a)} - a \right\} \geq 0,$$

using that $\varphi(a)/\overline{\Phi}(a) > a$ for all $a > 0$ [see e.g. 16]. For $a = 0$, we have that $\nu_a = \mathbb{E}(Z^2) = 1$, and the claim follows. \square

The following Lemma is due to Liu, Gao and Samworth [13].

Lemma F.2 (Liu, Gao and Samworth 13, Lemma 5). *Let $Z_i \stackrel{i.i.d.}{\sim} N(0, 1)$ for $i \in [p]$, where $p \in \mathbb{N}$. Let $a \geq 0$ and define $\nu_a = \mathbb{E}(Z^2 \mid |Z| \geq a)$. Then for all $x > 0$,*

$$\mathbb{P} \left[\sum_{i=1}^p (Z_i^2 - \nu_a) \mathbb{1}(|Z_i| \geq a) \geq 9 \left\{ (pe^{-a^2/2}x)^{1/2} + x \right\} \right] \leq e^{-x}.$$

The following Lemma is analogous to Lemma F.2, and gives a corresponding lower bound.

Lemma F.3. *Let $Z_i \stackrel{i.i.d.}{\sim} N(0, 1)$ for $i \in [p]$, where $p \in \mathbb{N}$. Let $a \geq 1$ and define $\nu_a = \mathbb{E}(Z_1^2 \mid |Z_1| \geq a)$. Then for all $x > 0$,*

$$\mathbb{P} \left[\sum_{i=1}^p (Z_i^2 - \nu_a) \mathbb{1}(|Z_i| \geq a) \leq -5 \left\{ (pe^{-a^2/2}x)^{1/2} + x \right\} \right] \leq e^{-x}.$$

Proof. The proof is similar to the proof of Lemma 5 in Liu, Gao and Samworth [13]. Let $X = (Z^2 - \nu_a) \mathbb{1}(|Z| \geq a)$, where $Z \sim N(0, 1)$. Let $\lambda \in (0, \frac{1}{2}]$. Then, as

$\mathbb{E}(X) = 0$, we have that

$$\mathbb{E}(e^{-\lambda X}) = 1 + \mathbb{E}(e^{-\lambda X} - 1 + \lambda X),$$

By the deterministic bound

$$e^{-y} - 1 + y \leq \begin{cases} y^2, & \text{if } y > 0, \\ y^2, & \text{if } -1 \leq y \leq 0, \\ e^{-y}, & \text{if } y \leq -1, \end{cases}$$

we obtain that

$$\begin{aligned} \mathbb{E}(e^{-\lambda X}) &\leq 1 + \lambda^2 \mathbb{E}\{X^2 \mathbb{1}(X > 0)\} + \lambda^2 \mathbb{E}\left\{X^2 \mathbb{1}\left(-\frac{1}{\lambda} \leq X \leq 0\right)\right\} \\ &\quad + \mathbb{E}\left\{e^{-\lambda X} \mathbb{1}\left(X < -\frac{1}{\lambda}\right)\right\}. \end{aligned}$$

We bound each term separately. Let $p(x)$ denote the density function of the χ_1^2 distribution. For the second term, we have that

$$\begin{aligned} \mathbb{E}\{X^2 \mathbb{1}(X > 0)\} &= \int_{\nu_a}^{\infty} (x - \nu_a)^2 p(x) dx \\ &= \int_{\nu_a}^{\infty} (x - \nu_a)^2 \frac{1}{(2\pi x)^{1/2}} e^{-x/2} dx \\ &\leq \frac{16}{(2\pi\nu_a)^{1/2}} e^{-\nu_a/2}, \end{aligned}$$

using that $1 + a^2 \leq \nu_a \leq a^2 + 2$ (Lemma F.1) and $a \geq 1$. For the third term, using that $X \geq a^2 - \nu_a \geq -2$ whenever $X \leq 0$, we have that

$$\begin{aligned} \mathbb{E}\left\{X^2 \mathbb{1}\left(-\frac{1}{\lambda} \leq X \leq 0\right)\right\} &\leq \mathbb{E}\left\{2^2 \mathbb{1}\left(-\frac{1}{\lambda} \leq X \leq 0\right)\right\} \\ &\leq 4\mathbb{E}\{\mathbb{1}(|Z| \geq a)\} \\ &\leq 8e^{-a^2/2}/(2\pi a^2)^{1/2} \\ &\leq 8e^{-a^2/2}/(2\pi)^{1/2}, \end{aligned}$$

where we in the penultimate step used the standard bound $\mathbb{P}(Z > a) \leq e^{-a^2/2}/(2\pi a^2)^{1/2}$ for all $a > 0$. For the last term, as $\lambda \leq 1/2$, we have that $\mathbb{P}(X < -\frac{1}{\lambda}) \leq \mathbb{P}(X < -2) = 0$, because $X \geq a^2 - \nu_a \geq -2$. Therefore, $\mathbb{E}\{e^{-\lambda X} \mathbb{1}(X < -1/\lambda)\} = 0$. Hence,

$$\begin{aligned} \mathbb{E}(e^{-\lambda X}) &\leq 1 + \lambda^2 \left\{ \frac{8}{(2\pi)^{1/2}} + \frac{16e^{-\frac{1}{2}}}{2\sqrt{\pi}} \right\} e^{-a^2/2} \\ &\leq 1 + 6\lambda^2 e^{-a^2/2} \end{aligned}$$

$$\leq \exp\left(6\lambda^2 e^{-a^2/2}\right).$$

By a Chernoff Bound we obtain that, for any $t > 0$,

$$\begin{aligned} \mathbb{P}\left\{\sum_{i=1}^n (Z_i^2 - \nu_a)\mathbb{1}(|Z_i| \geq a) < -t\right\} &= \mathbb{P}\left\{-\sum_{i=1}^n (Z_i^2 - \nu_a)\mathbb{1}(|Z_i| \geq a) > t\right\} \\ &\leq \inf_{0 < \lambda \leq \frac{1}{2}} e^{-\lambda t} \left\{\mathbb{E}\left(e^{-\lambda X}\right)\right\}^p \\ &\leq \inf_{0 < \lambda \leq \frac{1}{2}} \exp\left(-\lambda t + 6\lambda^2 p e^{-a^2/2}\right) \\ &\leq \exp\left\{-\left(\frac{t^2 e^{a^2/2}}{24p} \wedge \frac{t}{4}\right)\right\}. \end{aligned}$$

Now take $t = 5\{(pe^{-a^2/2}x)^{1/2} + x\}$ to obtain the result. □

The following Lemma is due to Birgé [2].

Lemma F.4 (Birgé 2, Lemma 8.1). *Let $Y \sim \chi_p^2(\Psi)$ have a non-central Chi Square distribution with p degrees of freedom and non-centrality parameter $\Psi \geq 0$. Then, for any $x > 0$, we have that*

$$\mathbb{P}\left[Y \geq p + \Psi + 2\{x(p + 2\Psi)\}^{1/2} + 2x\right] \leq e^{-x},$$

and,

$$\mathbb{P}\left[Y \leq p + \Psi - 2\{x(p + 2\Psi)\}^{1/2}\right] \leq e^{-x},$$

Lemma F.5. *Consider the model from Section 2, with one and only one changepoint η , and suppose $n \geq 3$ and $\sigma = 1$. Let $\mathcal{K} = \{i : \mu_{i,\eta+1} - \mu_{i,\eta} \neq 0\}$ denote the set of coordinates for which there is a change in mean, let $r(t)$ be defined as in (8), and let $h(t)$ be defined as in (10). Let the CUSUM transformation $T_{(s,e]}^v(\cdot)$ be defined as in (2), and for ease of notation, let $T^v(\cdot) = T_{(0,n]}^v(\cdot)$. Let $k = \|\mu_{\eta+1} - \mu_\eta\|_0$, and define $\beta_v = \sum_{i \in \mathcal{K}} \{T^\eta(\mu_{i,\cdot})^2 - T^v(\mu_{i,\cdot})^2\}$. Define the events*

$$\begin{aligned} \mathcal{E}_1 &= \left\{ \forall 0 < v < n, \sum_{i \in \mathcal{K}} \{C^\eta(i)^2 - C^v(i)^2\} \geq \beta_v - 2(2\beta_v \log n)^{1/2} - 16r(k) \right\}, \\ \mathcal{E}_2 &= \left\{ \forall 0 < v < n, \forall t \in \mathcal{T}, \sum_{i \in [p] \setminus \mathcal{K}} \{C^v(i)^2 - \nu_{a(t)}\} \mathbb{1}\{|C^v(i)| > a(t)\} \leq 63r(t) \right\} \\ \mathcal{E}_3 &= \left\{ \forall 0 < v < n, \forall t \in \mathcal{T}, \sum_{i \in [p] \setminus \mathcal{K}} \{C^v(i)^2 - \nu_{a(t)}\} \mathbb{1}\{|C^v(i)| > a(t)\} \geq -35r(t) \right\} \\ \mathcal{E}_4 &= \left\{ \forall 0 < v < n, \forall t \in \mathcal{T}, t < (p \log n)^{1/2} \right\}; \end{aligned}$$

$$\sum_{i=1}^p \left[\{C^v(i)^2 - \nu_{a(t)}\} \mathbb{1}\{|C^v(i)| > a(t)\} - C^v(i)^2 + 1 \right] \leq 5h(p) + 63r(t) \Bigg\}$$

Then $\mathbb{P}(\mathcal{E}_1 \cap \mathcal{E}_2 \cap \mathcal{E}_3 \cap \mathcal{E}_4) \geq 1 - \frac{1}{n}$.

Proof. By a union bound it suffices to consider each event separately.

Step 1. We first show that $\mathbb{P}(\mathcal{E}_1^c) \leq \frac{1}{3n}$. As the CUSUM is a linear operation and $X = \mu + W$, we have for any $0 < v < n$ and $i \in [p]$ that $C^v(i) = T^v(\mu_{i,\cdot}) + T^v(W_{i,\cdot})$. Hence, for any v , we have that

$$\begin{aligned} \sum_{i \in \mathcal{K}} \{C^\eta(i)^2 - C^v(i)^2\} &= \beta_v + \sum_{i \in \mathcal{K}} \{T^\eta(W_{i,\cdot})^2 - T^v(W_{i,\cdot})^2\} \\ &\quad + 2 \sum_{i \in \mathcal{K}} \{T^\eta(W_{i,\cdot})T^\eta(\mu_{i,\cdot}) - T^v(W_{i,\cdot})T^v(\mu_{i,\cdot})\}. \end{aligned}$$

We construct high-probability bounds on the two last terms separately. For the first term, note that since $W_{i,j} \stackrel{\text{i.i.d.}}{\sim} N(0, 1)$, for any fixed v we have $T^v(W_{i,\cdot}) \sim N(0, 1)$ independently for all $i \in [p]$. Hence $\sum_{i \in \mathcal{K}} T^\eta(W_{i,\cdot})^2 \sim \chi_k^2$ and $\sum_{i \in \mathcal{K}} T^v(W_{i,\cdot})^2 \sim \chi_k^2$. By Lemma F.4 and a union bound we therefore have that

$$\mathbb{P} \left[\sum_{i \in \mathcal{K}} \{T^\eta(W_{i,\cdot})^2 - T^v(W_{i,\cdot})^2\} \leq -4\{\log(9n^2)k\}^{1/2} - 2\log(9n^2) \right] \leq \frac{2}{9n^2}.$$

Using that $n \geq 3$ and the definition of $r(k)$, we obtain that

$$\mathbb{P} \left[\sum_{i \in \mathcal{K}} \{T^\eta(W_{i,\cdot})^2 - T^v(W_{i,\cdot})^2\} \leq -16r(k) \right] \leq \frac{2}{9n^2}. \tag{16}$$

To see this, consider first the case $k < (p \log n)^{1/2}$. Then $k \leq r(k)$ and $\log n \leq r(k)$, so $4\{\log(9n^2)k\}^{1/2} \leq 4\{4\log(n)k\}^{1/2} \leq 8r(k)$ and $2\log(9n^2) \leq 8\log n \leq 8r(k)$. For the case $k \geq (p \log n)^{1/2}$, we must have that $p \geq \log n$, and hence $4\{\log(9n^2)k\}^{1/2} \leq 8(p \log n)^{1/2} = 8r(k)$ and $2\log(9n^2) \leq 8\log(n) \leq 8(p \log n)^{1/2} = 8r(k)$.

For the second term, we make use of the fact that the CUSUM transformation $T^v(y)$ of any vector y can be expressed as an inner product. More precisely, define the n -dimensional vector $\Psi^v \in \mathbb{R}^n$ to have l th element given by

$$\Psi^v(l) = \begin{cases} \left(\frac{n-v}{nv}\right)^{1/2} & \text{for } l = 1, \dots, v, \\ -\left(\frac{v}{n(n-v)}\right)^{1/2} & \text{for } l = v + 1, \dots, n. \end{cases}$$

Then for any vector $y \in \mathbb{R}^n$, we have that

$$T^v(y) = \langle y, \Psi^v \rangle,$$

see Baranowski, Chen and Fryzlewicz [1]. Hence, for any $i \in \mathcal{K}$,

$$\begin{aligned} & T^\eta(\mu_{\cdot,i})T^\eta(W_{\cdot,i}) - T^v(\mu_{\cdot,i})T^v(W_{\cdot,i}) \\ &= \langle \mu_{i,\cdot}, \Psi^\eta \rangle \langle W_{i,\cdot}, \Psi^\eta \rangle - \langle \mu_{i,\cdot}, \Psi^v \rangle \langle W_{i,\cdot}, \Psi^v \rangle \\ &= \langle W_{i,\cdot}, \langle \mu_{i,\cdot}, \Psi^\eta \rangle \Psi^\eta \rangle - \langle W_{i,\cdot}, \langle \mu_{i,\cdot}, \Psi^v \rangle \Psi^v \rangle \\ &= \langle W_{i,\cdot}, \langle \mu_{i,\cdot}, \Psi^\eta \rangle \Psi^\eta - \langle \mu_{i,\cdot}, \Psi^v \rangle \Psi^v \rangle. \end{aligned}$$

As $W_{i,\cdot} \stackrel{\text{i.i.d.}}{\sim} N_n(0, I)$ for all $i \in \mathcal{K}$, we get that

$$T^\eta(\mu_{\cdot,i})T^\eta(W_{\cdot,i}) - T^v(\mu_{\cdot,i})T^v(W_{\cdot,i}) \stackrel{\text{i.i.d.}}{\sim} N\left(0, \|\langle \mu_{i,\cdot}, \Psi^\eta \rangle \Psi^\eta - \langle \mu_{i,\cdot}, \Psi^v \rangle \Psi^v\|_2^2\right),$$

for $i \in \mathcal{K}$. By Lemma F.11, we have that $\|\langle \mu_{i,\cdot}, \Psi^\eta \rangle \Psi^\eta - \langle \mu_{i,\cdot}, \Psi^v \rangle \Psi^v\|_2^2 = T^\eta(\mu_{i,\cdot})^2 - T^v(\mu_{i,\cdot})^2$. We therefore have that

$$\sum_{i \in \mathcal{K}} \{T^\eta(X_{i,\cdot})T^\eta(\mu_{i,\cdot}) - T^v(X_{i,\cdot})T^v(\mu_{i,\cdot})\} \sim N(0, \beta_v).$$

By the standard Gaussian tail bound $\mathbb{P}(Z > t) \leq e^{-t^2/2}$ for $Z \sim N(0, 1)$ and $t > 0$, we obtain

$$\mathbb{P}\left[\sum_{i \in \mathcal{K}} \{T^\eta(X_{i,\cdot})T^\eta(\mu_{i,\cdot}) - T^v(X_{i,\cdot})T^v(\mu_{i,\cdot})\} < -2(2\beta_v \log n)^{1/2}\right] \leq \frac{1}{9n^2}, \quad (17)$$

again using that $n \geq 3$. Combining (16) and (17) by a union bound, we have for any $0 < v < n$ that

$$\mathbb{P}\left[\sum_{i \in \mathcal{K}} \{C^\eta(i)^2 - C^v(i)^2\} \geq \beta_v - 2(2\beta_v \log n)^{1/2} - 16r(k)\right] \leq \frac{1}{3n^2}.$$

By another union bound (over v), we obtain that $\mathbb{P}(\mathcal{E}_1^c) \leq \frac{1}{3n}$.

Step 2. We now show that $\mathbb{P}(\mathcal{E}_2^c) \leq 1/6n$. Fix $0 < v < n$ and any $t \in \mathcal{T}$ such that $t \leq (p \log n)^{1/2}$. Fix $x_t > 0$, to be determined later. As $C^v(i) \stackrel{\text{i.i.d.}}{\sim} N(0, 1)$ for all $i \in \mathcal{K}^c$, we have by Lemma F.2 that

$$\sum_{i \in \mathcal{K}^c} \{C^v(i)^2 - \nu_{a(t)}\} \mathbb{1}\{|C^v(i)| > a(t)\} \leq 9 \left[\left\{ p e^{-a(t)^2/2} x_t \right\}^{1/2} + x_t \right] \quad (18)$$

with probability at least $1 - e^{-x_t}$. By a union bound, (18) holds for all $0 < v < n$ and $t \in \mathcal{T}$ such that $t \leq (p \log n)^{1/2}$, with probability at least $1 - n \sum_{t \in \mathcal{T} \setminus \{p\}} e^{-x_t}$. Now set $x_t = 6 \{p \log^2(n)/t^2 \wedge r(t)\}$. Then

$$\sum_{t \in \mathcal{T} \setminus \{p\}} e^{-x_t} \leq \sum_{t \in \mathcal{T} \setminus \{p\}} \exp\left\{-6 \frac{p \log^2(n)}{t^2}\right\} + \sum_{t \in \mathcal{T} \setminus \{p\}} \exp\{-6r(t)\}.$$

For the first sum, we have

$$\begin{aligned} \sum_{t \in \mathcal{T} \setminus \{p\}} \exp \left\{ -6 \frac{p \log^2(n)}{t^2} \right\} &\leq \sum_{k=0}^{\infty} \exp \{ -6 \log(n) 4^k \} \\ &= \sum_{k=0}^{\infty} \left(\frac{1}{n^6} \right)^{4^k} \\ &\leq \frac{1}{n^6} + \frac{1}{n^6} \sum_{k=1}^{\infty} \left(\frac{1}{n^6} \right)^{3k} \\ &= \frac{1}{n^6} \left(1 + \frac{1}{n^{18} - 1} \right). \end{aligned}$$

For the second sum, noting that $6r(t) = 6 \{ t \log(ep \log n/t^2) \vee \log n \} \geq 3t \log(ep \log n/t^2) + 3 \log n$, we have

$$\begin{aligned} \sum_{t \in \mathcal{T} \setminus \{p\}} \exp \{ -6r(t) \} &\leq \frac{1}{n^3} \sum_{t \in \mathcal{T} \setminus \{p\}} \left(\frac{t^2}{ep \log n} \right)^{3t} \\ &\leq \frac{1}{n^3 e^3} \left(1 + \sum_{k=1}^{\infty} 4^{-3k} \right). \end{aligned}$$

With our choice of x_t , using that $a^2(t) = 4 \log(ep \log n/t^2)$, we have that

$$\begin{aligned} 9 \left[\left\{ p e^{-a^2(t)/2} x_t \right\}^{1/2} + x_t \right] &= 9 \left[\left\{ p \frac{t^4}{e^2 p^2 \log^2 n} x_t \right\}^{1/2} + x_t \right] \\ &\leq 9 \left\{ \frac{t\sqrt{6}}{e} + 6r(t) \right\} \\ &\leq 63r(t), \end{aligned}$$

where we used that $x_t \leq 6r(t)$ and $x_t \leq 6p \log^2(n)/t^2$, as well as the fact that $t \leq r(t)$ whenever $t \leq (p \log n)^{1/2}$. Hence, using that $n \geq 3$,

$$\begin{aligned} &\mathbb{P} \left[\exists 0 < v < n, \exists t \in \mathcal{T}, t \leq (p \log n)^{1/2}; \right. \\ &\quad \left. \sum_{i \in \mathcal{K}^c} \{ C^v(i)^2 - \nu_{a(t)} \} \mathbb{1}_{\{|C^v(i)| > a(t)\}} > 63r(t) \right] \\ &\leq \frac{1}{n^6} \left(1 + \frac{1}{n^{18} - 1} \right) + \frac{1}{n^3 e^3} \left(1 + \sum_{k=1}^{\infty} 4^{-3k} \right) \\ &\leq \frac{1}{18n^2}. \end{aligned} \tag{19}$$

Now consider the case where $t = p$. If $p \leq (p \log n)^{1/2}$, then $a(p) > 0$ and similarly as above we have that

$$\begin{aligned} & \mathbb{P} \left[\exists 0 < v < n ; \sum_{i \in \mathcal{K}^c} \{C^v(i)^2 - \nu_{a(p)}\} \mathbb{1}\{|C^v(i)| > a(p)\} > 63r(p) \right] \\ & \leq \frac{1}{18n^2}. \end{aligned} \quad (20)$$

If we instead have $p > (p \log n)^{1/2}$, in which case $a(p) = 0$ and $\nu_{a(p)} = 1$, then for any $0 < v < n$, we have

$$\sum_{i \in \mathcal{K}^c} \{C^v(i)^2 - \nu_{c(p)}\} \mathbb{1}\{|C^v(i)| > c(p)\} = \sum_{i \in \mathcal{K}^c} \{C^v(i)^2\} - p + k.$$

As $\sum_{i \in \mathcal{K}^c} C^v(i)^2 \sim \chi_{p-k}^2$, we obtain from Lemma F.4 that

$$\sum_{i \in \mathcal{K}^c} \{C^v(i)^2\} - p + k > 2 \{p \log(12n^2)\}^{1/2} + 2 \log(12n^2),$$

with probability at most $1/(12n^2)$. Using that $n \geq 3$ and $r(p) \geq \log n$, we obtain by a union bound that

$$\mathbb{P} \left[\exists 0 < v < n ; \sum_{i \in \mathcal{K}^c} \{C^v(i)^2 - \nu_{c(p)}\} \mathbb{1}\{|C^v(i)| > c(p)\} > 15r(t) \right] \leq \frac{1}{12n}. \quad (21)$$

Combining (19), (20) and (21) by a union bound, we have that $\mathbb{P}(\mathcal{E}_2^c) \leq 1/(6n)$.

Step 3. We show that $\mathbb{P}(\mathcal{E}_3^c) \leq 1/6n$. Fix $0 < v < n$ and any $t \in \mathcal{T}$ such that $t \leq (p \log n)^{1/2}$. Fix $x_t > 0$, to be determined later. As $C^v(i) \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$ for all $i \in \mathcal{K}^c$, we have by Lemma F.3 that

$$\sum_{i \in \mathcal{K}^c} \{C^v(i)^2 - \nu_{a(t)}\} \mathbb{1}\{|C^v(i)| > a(t)\} \geq -5 \left[\left\{ p e^{-a(t)^2/2} x_t \right\}^{1/2} + x_t \right] \quad (22)$$

with probability at least $1 - e^{-x_t}$. By a union bound, (22) holds for all $0 < v < n$ and $t \in \mathcal{T}$ such that $t \leq (p \log n)^{1/2}$, with probability at least $1 - n \sum_{t \in \mathcal{T} \setminus \{p\}} e^{-x_t}$. Now set $x_t = 6 \{p \log^2(n)/t^2 \wedge r(t)\}$. Similar to Step 2, we obtain that

$$\begin{aligned} & \mathbb{P} \left[\exists 0 < v < n, \exists t \in \mathcal{T}, t \leq (p \log n)^{1/2} ; \right. \\ & \quad \left. \sum_{i \in \mathcal{K}^c} \{C^v(i)^2 - \nu_{a(t)}\} \mathbb{1}\{|C^v(i)| > a(t)\} < -35r(t) \right] \\ & \leq \frac{1}{18n^2}. \end{aligned}$$

Also similar to Step 2, we have that

$$\begin{aligned} & \mathbb{P} \left[\exists 0 < v < n ; \sum_{i \in \mathcal{K}^c} \{C^v(i)^2 - \nu_{a(p)}\} \mathbb{1}\{|C^v(i)| > a(p)\} < -35r(p) \right] \\ & \leq \frac{1}{12n^2}. \end{aligned}$$

It then follows that $\mathbb{P}(\mathcal{E}_3^c) \leq 1/(6n)$ by a union bound.

Step 4. Lastly we show that $\mathbb{P}(\mathcal{E}_4^c) \leq 1/(3n)$. Fix any $0 < v < n$ and $t \in \mathcal{T}$ such that $t < (p \log n)^{1/2}$. By Lemma F.8 and Theorem 1.A.3(b) in Shaked and Shanthikumar [17], we have that

$$\begin{aligned} & \sum_{i=1}^p [\{C^v(i)^2 - \nu_{a(t)}\} \mathbb{1}\{|C^v(i)| > a(t)\} - C^v(i)^2 + 1] \\ & \leq_{\text{st}} \sum_{i=1}^p [\{Y_i^2 - \nu_{a(t)}\} \mathbb{1}\{|Y_i| > a(t)\} - Y_i^2 + 1], \end{aligned}$$

where $Y_i \stackrel{\text{i.i.d.}}{\sim} N(0, 1)$ for $i \in [p]$. Hence,

$$\begin{aligned} \mathbb{P}(\mathcal{E}_4^c) & \leq \sum_{0 < v < n} \sum_{t \in \mathcal{T} \setminus \{p\}} \mathbb{P} \left[\sum_{i=1}^p \{Y_i^2 - \nu_{a(t)}\} \mathbb{1}\{|Y_i| > a(t)\} \geq 63r(t) \right] \\ & \quad + \sum_{0 < v < n} \sum_{t \in \mathcal{T} \setminus \{p\}} \mathbb{P} \left\{ \sum_{i=1}^p \{Y_i^2 - 1\} \leq -5h(p) \right\} \\ & \leq \frac{1}{18n} + n \log_2(p) \mathbb{P} \left\{ \sum_{i=1}^p (Y_i^2 - 1) \leq -5h(p) \right\}, \end{aligned}$$

where we for the first sum used the same arguments as in Step 2. For the second sum, we have by Lemma F.4 that

$$\mathbb{P} \left[\sum_{i=1}^p (Y_i^2 - 1) \leq -2p^{1/2} \{\log(6n^2) + \log \log_2 p\}^{1/2} \right] \leq \frac{1}{6n^2 \log_2 p}.$$

Now,

$$\begin{aligned} 2p^{1/2} \{\log(6n^2) + \log \log_2 p\}^{1/2} & \leq 2 [6 \{\log n \vee \log \log(ep)\}]^{1/2} \\ & \leq 5h(p). \end{aligned}$$

Hence $\mathbb{P}(\mathcal{E}_4^c) \leq 1/(18n) + 1/(6n) \leq 1/(3n)$, and the proof is complete. \square

The following Lemma gives high-probability control over the penalized score $S_{\gamma, (s, e]}^v$ in (6) used as a test statistic.

Lemma F.6. Consider the model from Section 2, and assume $\sigma = 1$. Let $r(t)$ be defined as in (8). For any integer v such that $s < v < e$, let $T_{(s,e]}^v(\cdot)$ be defined as in (2), and define

$$\beta_{(s,e]}^v = \sum_{i=1}^p T_{(s,e]}^v(\mu_{i,\cdot})^2,$$

and

$$k_{(s,e]} = \sum_{i=1}^p \mathbb{1} \left\{ T_{(s,e]}^v(\mu_{i,\cdot})^2 = 0 \right\}.$$

Note that if $\beta_{(s,e]}^v = 0$ for some v , then the open integer interval (s, e) contains no changepoint. Define the events

$$\begin{aligned} \mathcal{E}_5 &:= \left\{ \forall 0 \leq s < v < e \leq n, \beta_{(s,e]}^v = 0 ; S_{\gamma,(s,e]}^v < 0 \right\}, \\ \mathcal{E}_6 &:= \left\{ \forall 0 \leq s < v < e \leq n ; S_{\gamma,(s,e]}^v \geq \beta_{(s,e]} - 8 \left\{ 2\beta_{(s,e]}^v r(k_{(s,e]}) \right\}^{1/2} \right. \\ &\quad \left. - (\gamma + 106) r(k_{(s,e]}) \right\}. \end{aligned}$$

If $\gamma \geq 82$, then $\mathbb{P}(\mathcal{E}_5 \cap \mathcal{E}_6) \geq 1 - 1/n$.

Proof.

Step 1. We first show that $\mathbb{P}(\mathcal{E}_5^c) \leq 1/(2n)$. Consider any integer triple of s, e, v such that $0 \leq s < v < e \leq n$ and $\beta_{(s,e]}^v = 0$. Fix any $t \in \mathcal{T} \setminus \{p\}$ (the case $t = p$ is handled later), and fix $x_t > 0$, to be specified later. As $\beta_{(s,e]}^v = 0$, the open integer interval (s, e) contains no changepoint, and thus $C_{(s,e]}^v(i) \stackrel{\text{i.i.d.}}{\sim} N(0, 1)$ for all $i \in [p]$. By Lemma F.2 we have that

$$\sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(t)} \right\} \mathbb{1} \left\{ \left| C_{(s,e]}^v(i) \right| > a(t) \right\} \geq 9 \left[\left\{ p e^{-a(t)^2/2} x_t \right\}^{1/2} + x_t \right] \tag{23}$$

occurs with probability at most e^{-x_t} . Note that there are at most n^3 unique choices of the triple (s, e, v) . By a union bound, (23) holds for some $0 \leq s < v < e \leq n$ and some $t \in \mathcal{T} \setminus \{p\}$ with probability at most $n^3 \sum_{t \in \mathcal{T} \setminus \{p\}} e^{-x_t}$. Now set $x_t = 8 \left\{ \frac{p \log^2(n)}{t^2} \wedge r(t) \right\}$ for all t . Then,

$$\sum_{t \in \mathcal{T} \setminus \{p\}} e^{-x_t} \leq \sum_{t \in \mathcal{T} \setminus \{p\}} \exp \left\{ -8 \frac{p \log^2(n)}{t^2} \right\} + \sum_{t \in \mathcal{T} \setminus \{p\}} \exp \{-8r(t)\}.$$

For the first sum, we have

$$\sum_{t \in \mathcal{T} \setminus \{p\}} \exp \left\{ -8 \frac{p \log^2(n)}{t^2} \right\} \leq \sum_{k=0}^{\infty} \exp \{-8 \log(n) 4^k\}$$

$$\begin{aligned} &= \sum_{k=0}^{\infty} \left(\frac{1}{n^8}\right)^{4k} \\ &\leq \frac{1}{n^8} + \frac{1}{n^8} \sum_{k=1}^{\infty} \left(\frac{1}{n^8}\right)^{3k} \\ &= \frac{1}{n^8} \left(1 + \frac{1}{n^{24} - 1}\right). \end{aligned}$$

For the second sum, noting that $8r(t) = 8\{t \log(\frac{ep \log n}{t^2}) \vee \log n\} \geq 4t \log(\frac{ep \log n}{t^2}) + 4 \log n$, we have

$$\begin{aligned} \sum_{t \in \mathcal{T} \setminus \{p\}} \exp\{-8r(t)\} &\leq \frac{1}{n^4} \sum_{t \in \mathcal{T} \setminus \{p\}} \left(\frac{t^2}{ep \log n}\right)^{4t} \\ &\leq \frac{1}{n^4 e^4} \left(1 + \sum_{k=1}^{\infty} 4^{-4k}\right). \end{aligned}$$

Hence, using that $n \geq 2$,

$$\begin{aligned} n^3 \sum_{t \in \mathcal{T} \setminus \{p\}} e^{-x_t} &\leq \frac{1}{n^5} \left(1 + \frac{1}{n^{24} - 1}\right) + \frac{1}{ne^4} \left(1 + \sum_{k=1}^{\infty} 4^{-4k}\right) \\ &\leq \frac{1}{10n}. \end{aligned}$$

With this choice of x_t , using that $a^2(t) = 4 \log\left(\frac{ep \log n}{t^2}\right)$, we have that

$$\begin{aligned} 9 \left[\left\{ p e^{-c^2(t)/2} x_t \right\}^{1/2} + x_t \right] &= 9 \left[\left\{ p \frac{t^4}{e^2 p^2 \log^2 n} x_t \right\}^{1/2} + x_t \right] \\ &\leq 9 \left\{ \frac{t\sqrt{8}}{e} + 8r(t) \right\} \\ &\leq 82r(t), \end{aligned}$$

where we used that $x_t \leq 8r(t)$ and $x_t \leq 8p \log^2(n)/t^2$, as well as the fact that $t \leq r(t)$ whenever $t \leq (p \log n)^{1/2}$. Hence,

$$\begin{aligned} &\mathbb{P} \left[\exists 0 \leq s < v < e \leq n, \exists t \in \mathcal{T} \setminus \{p\}; \right. \\ &\quad \left. \sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(t)} \right\} \mathbb{1} \left\{ |C_{(s,e]}^v(i)| > a(t) \right\} \geq 82r(t) \right] \\ &\leq \frac{1}{10n}. \end{aligned} \tag{24}$$

Now consider the case where $t = p$. If $p \leq (p \log n)^{1/2}$, then similarly as above we have that

$$\begin{aligned} & \mathbb{P} \left[\exists 0 \leq s < v < e \leq n, ; \right. \\ & \quad \left. \sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(p)} \right\} \mathbb{1} \left\{ |C_{(s,e]}^v(i)| > a(p) \right\} \geq 82r(p) \right] \\ & \leq \frac{1}{10n}. \end{aligned} \quad (25)$$

If we instead have $p > (p \log n)^{1/2}$ (in which case $a(p) = 0$ and $\nu_{a(p)} = 1$), then for any $0 \leq s < v < e \leq n$, we have

$$\sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{c(p)} \right\} \mathbb{1} \left\{ |C_{(s,e]}^v(i)| > c(p) \right\} = \sum_{i=1}^p C_{(s,e]}^v(i)^2 - p.$$

As $\sum_{i=1}^p C_{(s,e]}^v(i)^2 \sim \chi_p^2$, we obtain from Lemma F.4 that

$$\sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 \right\} - p \geq 2 \{p \log(4n^4)\}^{1/2} + 2 \log(4n^4),$$

occurs with probability at most $1/(4n^4)$. Using that $n \geq 2$ and $r(p) \geq \log n$, we obtain by a union bound that

$$\begin{aligned} & \mathbb{P} \left[\exists 0 \leq s < v < e \leq n ; \sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{c(p)} \right\} \mathbb{1} \left\{ |C_{(s,e]}^v(i)| > c(p) \right\} \geq 17r(t) \right] \\ & \leq \frac{1}{4n}, \end{aligned} \quad (26)$$

Combining (24), (25) and (26) by a union bound, and using that $\gamma \geq 82$, we get that $\mathbb{P}(\mathcal{E}_5^c) \leq 1/(2n)$.

Step 2. Now we show that $\mathbb{P}(\mathcal{E}_6^c) \leq 1/(2n)$. Consider any $0 \leq s < v < e \leq n$. Without loss of generality, assume that $T_{(s,e]}^v(\mu_{1,\cdot})^2 \geq T_{(s,e]}^v(\mu_{2,\cdot})^2 \geq \dots \geq T_{(s,e]}^v(\mu_{p,\cdot})^2$. Let z denote the smallest integer in \mathcal{T} no smaller than $k_{(s,e]}$, where we suppress the dependence of z on s, e in the notation. For $z \leq (p \log n)^{1/2}$, observe that

$$\begin{aligned} & \sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(z)} \right\} \mathbb{1} \left\{ |C_{(s,e]}^v(i)| > a(z) \right\} \\ & \geq \sum_{i=1}^{k_{(s,e]}} \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(z)} \right\} + \sum_{i=k_{(s,e]}+1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(z)} \right\} \mathbb{1} \left\{ |C_{(s,e]}^v(i)| > a(z) \right\}. \end{aligned} \quad (27)$$

We lower bound the two sums separately. For each $i \in [p]$ we have that $C_{(s,e]}^v = T_{(s,e]}^v(\mu_{i,\cdot}) + T_{(s,e]}^v(W_{i,\cdot}) \stackrel{\text{ind}}{\sim} N(T_{(s,e]}^v(\mu_{i,\cdot}), 1)$. Let $x_t = 8 \{p \log^2(n)/t^2 \wedge r(t)\}$ for all $t \in [p]$, as in Step 1. For the first sum, noting that $\sum_{i=1}^{k_{(s,e]}} C_{(s,e]}^v(i)^2 \sim \chi_{k_{(s,e]}}^2(\beta_{(s,e]}^v)$ (a non-central Chi Square distribution with $k_{(s,e]}$ degrees of freedom and non-centrality parameter $\beta_{(s,e]}^v$), we have by Lemma F.4 that

$$\mathbb{P} \left[\sum_{i=1}^{k_{(s,e]}} \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(z)} \right\} < k_{(s,e]} - k_{(s,e]}\nu_{a(z)} + \beta_{(s,e]}^v - 2 \left\{ x_z \left(z + 2\beta_{(s,e]}^v \right) \right\}^{1/2} \right] \leq e^{-x_z}.$$

Note that, since $k_{(s,e]} \leq z \leq (p \log n)^{1/2}$, we have $z \leq r(z)$ and $k_{(s,e]} \leq r(k_{(s,e]})$. Moreover, by Lemma F.1, we have $\nu_{a(z)}^2 \leq 2 + a^2(z) \leq 2 + a^2(k_{(s,e]})$, where we for the last inequality used that $z \geq k_{(s,e]}$ and that $t \mapsto a^2(t)$ is decreasing. Since $z \leq 2k_{(s,e]}$, it also holds that $r(z) \leq 2r(k_{(s,e]})$. Hence,

$$\mathbb{P} \left[\sum_{i=1}^{k_{(s,e]}} \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(z)} \right\} < \beta_{(s,e]}^v - 8 \left\{ 2\beta_{(s,e]}^v r(k_{(s,e]}) \right\} - 14r(k_{(s,e]}) \right] \leq e^{-x_z}.$$

For the second sum, we obtain from Lemma F.3 that

$$\mathbb{P} \left[\sum_{i=k_{(s,e]}+1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(z)} \right\} \mathbb{1} \left\{ |C_{(s,e]}^v(i)| > a(z) \right\} \leq -5 \left[\left\{ p e^{-b^2(z)/2} x_z \right\}^{1/2} + x_z \right] \right] \leq e^{-x_t}.$$

By the definition of x_z , we have that

$$5 \left[\left\{ p e^{-b^2(z)/2} x_z \right\}^{1/2} + x_z \right] \leq 46r(z) \leq 92r(k_{(s,e]}).$$

By a union bound over the two sums in (27), we have that

$$\sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(z)} \right\} \mathbb{1} \left\{ |C_{(s,e]}^v(i)| > a(z) \right\} < \beta_{(s,e]}^v - 8 \left\{ 2\beta_{(s,e]}^v(z) r(k_{(s,e]}) \right\}^{1/2} - 106r(k_{(s,e]}) \tag{28}$$

occurs with probability at most $2e^{-x_t}$.

Now suppose that $z > (p \log n)^{1/2}$. Then,

$$\sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 - \nu_{a(z)} \right\} \mathbb{1} \left\{ |C_{(s,e]}^v(i)| > a(z) \right\} = \sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 \right\} - p.$$

Using that $\sum_{i=1}^p C_{(s,e]}^v(i)^2 \sim \chi_p^2(\beta_{(s,e]}^v)$, we have by Lemma F.4 that

$$\sum_{i=1}^p \left\{ C_{(s,e]}^v(i)^2 \right\} - p < \beta_{(s,e]}^v - 2 \left\{ \log(4n^4)(p + 2\beta_{(s,e]}^v) \right\}^{1/2}$$

occurs with probability at most $1/(4n^4)$. In particular, since $\log n \leq r(t)$ for all t , we have $r(z) = r(\sqrt{p \log n}) \leq 2r(k_{(s,e]})$ and $n \geq 2$, this implies that (28) occurs probability at most $1/(4n^4)$ whenever $z \geq (p \log n)^{1/2}$. By a union bound over $0 \leq s < v < e \leq n$, we obtain that

$$\begin{aligned} \mathbb{P}(\mathcal{E}_6^c) &\leq n^3 \left(\frac{1}{4n^4} + 2 \sum_{t \in \mathcal{T} \setminus \{p\}} e^{-x_t} \right) \\ &\leq \frac{1}{4n} + \frac{1}{5n} \\ &\leq \frac{1}{2n}, \end{aligned}$$

where we used the same approach as in Step 1 to bound $\sum_{t \in \mathcal{T} \setminus \{p\}} e^{-x_t}$. The proof is complete. \square

Lemma F.7. *Let \mathcal{M} denote the collection of seeded intervals generated by Algorithm 4 with parameters $\alpha \in (1, 2]$ and $K \geq 2$. Then for all real numbers $h > 0$ such that $h \leq n/2$, and all integers η such that $3h/2 \vee 1 \leq \eta \leq n - (3h/2 \vee 1)$, there exists integers $l \geq 1$ and v such that the following holds.*

- (P1) $(v - l, v + l] \in \mathcal{M}$;
- (P2) $h/2 \leq l \leq h \vee 1$;
- (P3) $|v - \eta| \leq l/K \leq l/2$.

In particular, $(v - l, v + l] \subseteq (\eta - (3/2h \vee 1), \eta + (3/2h \vee 1)]$.

Proof. Define the recursive sequence $(l_j)_{j \in \mathbb{N}}$ by $l_1 = 1$, and $l_{j+1} = \max\{l_j + 1, \lfloor \alpha l_j \rfloor\}$ for $j \in \mathbb{N}$. Let $H = \max\{j \in \mathbb{N} : l_j \leq n/2\}$. Formally, the set \mathcal{M} of seeded intervals generated by Algorithm 4 is given by

$$\mathcal{S} = \bigcup_{l \in \{l_1, \dots, l_H\}} \mathcal{I}_l,$$

where

$$\mathcal{I}_l = \{(n - 2l, n]\} \cup \bigcup_{i=0}^{\lfloor \frac{n-2l}{s_l} \rfloor} \{(is_l, is_l + 2l]\},$$

$$s_l = \max \left\{ 1, \left\lfloor \frac{l}{K} \right\rfloor \right\}.$$

Note that, for any $j \in [H - 1]$, it holds that $l_{j+1}/l_j \leq \max\{2, \alpha\} = 2$. Hence, there must exist an integer $j \in [H]$ such that $h/2 \leq l_j \leq h \vee 1$. Moreover, by the definition of \mathcal{I}_{l_j} , there must exist an integer v such that $|v - \eta| \leq \lfloor l_j/K \rfloor \leq l_j/2$ and $(v - l_j, v + l_j) \in \mathcal{I}_{l_j}$. This proves the first three claims. For the last claim, note that

$$\begin{aligned} v - l_j &= v - \eta + \eta - l_j \\ &\geq -\lfloor l_j/K \rfloor + \eta - l_j \\ &\geq \eta - (3/2h \vee 1). \end{aligned}$$

Similarly, we have that $v + l_j \leq \eta + (3/2h \vee 1)$. □

Lemma F.8. *Let $Y \sim N(\theta, 1)$, $\theta \in \mathbb{R}$, $a > 0$ and $\nu_a = \mathbb{E}(Y^2 \mid |Y| \geq a)$. Let $A = (Y^2 - \nu_a) \mathbb{1}(|Y| \geq a)$ and $B = Y^2 - 1$. Then $A - B$ is stochastically decreasing in $|\theta|$.*

Proof. It is equivalent to show that $B - A$ is stochastically increasing in $|\theta|$. Note first that Y^2 has a Chi Square distribution with non-centrality parameter θ^2 , which is stochastically increasing in $|\theta|$. Further, we have that $B - A = f(Y^2)$, where the function f is given by

$$f(x) = \begin{cases} x - 1, & \text{if } x < a^2 \\ \nu_a - 1, & \text{otherwise.} \end{cases}$$

Since $\nu_a \geq a^2$, f is an increasing function. By Shaked and Shanthikumar [17, Theorem 1.A.3(a)], $B - A$ must be stochastically increasing in $|\theta|$. □

Lemma F.9. *Let \mathcal{M} denote the set of candidate intervals generated from Algorithm 4 with fixed input parameters $\alpha > 1$, $K > 1$ and $n \in \mathbb{N}$. Then the number of distinct triples of integers s, e, v such that $(s, e) \in \mathcal{M}$ and $s < v < e$ is of order $\mathcal{O}(n \log n)$.*

Proof. Let α and K be given, and define the recursive sequence $(l_j)_{j \in \mathbb{N}}$ by $l_1 = 1$, and $l_{j+1} = \max(l_j + 1, \lfloor \alpha l_j \rfloor)$ for $j \in \mathbb{N}$. Let $H = \sup\{j \in \mathbb{N} : l_j \leq n/2\}$. Formally, the set \mathcal{M} of seeded intervals generated by Algorithm 4 is given by

$$\mathcal{S} = \bigcup_{l \in \{l_1, \dots, l_H\}} \mathcal{I}_l,$$

where

$$\mathcal{I}_l = \{(n - 2l, n]\} \cup \bigcup_{i=0}^{\lfloor \frac{n-2l}{s_l} \rfloor} \{(is_l, is_l + 2l]\},$$

and

$$s_l = \max \left\{ 1, \left\lfloor \frac{l}{K} \right\rfloor \right\}.$$

For any $(s, e] \in \mathcal{I}_l$, there are precisely $2l - 1 < 2l$ integers v such that $s < e < v$. Hence, the number N of distinct triples of integers s, e, v such that $(s, e] \in \mathcal{M}$ and $s < v < e$ therefore satisfies

$$N < \sum_{l \in \{l_1, \dots, l_J\}} 2l |\mathcal{I}_l|.$$

For all $l < K$, we have that $|\mathcal{I}_l| \leq n$, and so $2l|\mathcal{I}_l| \leq 2ln < 2Kn$. For $l \geq K$ we have that $\lfloor l/K \rfloor \geq l/(2K)$, and so $2l|\mathcal{I}_l| \leq 4Kn$. Therefore,

$$\begin{aligned} N &< \sum_{l \in \{l_1, \dots, l_J\}} 4Kn \\ &= 4HKn. \end{aligned}$$

Noting that $H \leq \lceil \frac{1}{\alpha-1} \rceil + \log_\alpha n$, we thus get

$$\begin{aligned} N &< 4 \left(\lceil \frac{1}{\alpha-1} \rceil + \log_\alpha n \right) Kn \\ &= \mathcal{O}(n \log n), \end{aligned}$$

which gives the desired result. □

In the following we restate some useful Lemmas from Baranowski, Chen and Fryzlewicz [1].

Lemma F.10 (Baranowski, Chen and Fryzlewicz 1, Lemma 2).

Consider the model from Section 2, assuming that $p = 1$. Let the CUSUM transformation $T_{(s,e]}^v(\cdot)$ be defined as in (2). Suppose $s < e$ are such that $\eta_{j-1} \leq s < \eta_j < e \leq \eta_{j+1}$ for some $j \in [J]$. Then,

$$\max_{s < v < e} T_{(s,e]}^v(\mu)^2 = T_{(s,e]}^{\eta_j}(\mu)^2 \begin{cases} \geq \frac{1}{2} \Delta_j \theta_j^2 \\ \leq \Delta_j \theta_j^2. \end{cases}$$

Given an $n \in \mathbb{N}$ and any integer $0 < v < n$, define the n -dimensional vector $\Psi^v \in \mathbb{R}^n$ to have l th element given by

$$\Psi^v(l) = \begin{cases} \left(\frac{n-v}{nv} \right)^{1/2}, & \text{for } l = 1, \dots, v, \\ - \left(\frac{v}{n(n-v)} \right)^{1/2}, & \text{for } l = v + 1, \dots, n. \end{cases}$$

Lemma F.11 (Baranowski, Chen and Fryzlewicz 1, Lemma 4). Consider the model from Section 2, assuming that $p = 1$. Let the CUSUM transformation

$T_{(s,e]}^v(\cdot)$ be defined as in (2). Pick any interval $(s, e] \subset (0, n]$ such that the open integer interval (s, e) contains precisely one changepoint η_j . Pick any integer v such that $s < v < e$. Define $\rho = |\eta_j - v|$, $\delta_L = \eta_j - s$, and $\delta_R = e - \eta_j$. Then,

$$\left\| \Psi_{(s,e]}^{\eta_j} \langle \mu, \Psi_{(s,e]}^{\eta_j} \rangle - \Psi_{(s,e]}^v \langle \mu, \Psi_{(s,e]}^v \rangle \right\|_2 = T_{(s,e]}^{\eta_j}(\mu)^2 - T_{(s,e]}^v(\mu)^2.$$

Moreover,

$$(1) \text{ for any } \eta_j \leq v < e, \quad T_{(s,e]}^{\eta_j}(\mu)^2 - T_{(s,e]}^v(\mu)^2 = \frac{\rho \delta_L}{\rho + \delta_L} \theta_j^2;$$

$$(2) \text{ for any } s < v \leq \eta_j, \quad T_{(s,e]}^{\eta_j}(\mu)^2 - T_{(s,e]}^v(\mu)^2 = \frac{\rho \delta_R}{\rho + \delta_R} \theta_j^2.$$

Acknowledgments

We thank Idris Eckley, Arnaldo Frigessi and Nils Lid Hjort for constructive discussions and feedback, and the two anonymous referees whose comments greatly improved this paper. We also thank Camilla Feurst and Statkraft for providing the data used in our real-life example, and Abhishek Kaul for sharing his R code to be used in our simulation studies.

Funding

This project has been partially funded by the centers BigInsight, Norwegian Research Council, project number 237718, and Integreat, Norwegian Research Council, project number 332645

References

- [1] BARANOWSKI, R., CHEN, Y. and FRYZLEWICZ, P. (2019). Narrowest-over-threshold detection of multiple change points and change-point-like features. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **81** 649–672. <https://doi.org/10.1111/rssb.12322>. MR3961502
- [2] BIRGÉ, L. (2001). An alternative point of view on Lepski's method. *State of the art in probability and statistics* **36** 113–134. <https://doi.org/10.1214/lnms/1215090065>. MR1836557
- [3] CHO, H. (2016). Change-point detection in panel data via double CUSUM statistic. *Electronic Journal of Statistics* **10** 2000–2038. <https://doi.org/10.1214/16-EJS1155>. MR3522667
- [4] CHO, H. and FRYZLEWICZ, P. (2015). Multiple-change-point detection for high dimensional time series via sparsified binary segmentation. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* **77** 475–507. <https://doi.org/10.1111/rssb.12079>. MR3310536
- [5] CHO, H. and FRYZLEWICZ, P. (2018). *hdbinseg: Change-Point Analysis of High-Dimensional Time Series via Binary Segmentation*. CRAN. <https://CRAN.R-project.org/package=hdbinseg>.

- [6] CUNEN, C., HJORT, N. L. and NYGÅRD, H. M. (2020). Statistical sightings of better angels: Analysing the distribution of battle-deaths in interstate conflict over time. *Journal of Peace Research* **57** 221–234. <https://doi.org/10.1177/0022343319896843>
- [7] FRYZLEWICZ, P. (2014). Wild binary segmentation for multiple changepoint detection. *The Annals of Statistics* **42** 2243–2281. <https://doi.org/10.1214/14-AOS1245>. MR3269979
- [8] GAO, Z., DU, P., JIN, R. and ROBERTSON, J. L. (2020). Surface temperature monitoring in liver procurement via functional variance changepoint analysis. *The Annals of Applied Statistics* **14** 143–159. <https://doi.org/10.1214/19-AOAS1297>. MR4085087
- [9] KAUL, A. and MICHAILEDIS, G. (2024). Inference for change points in high dimensional mean shift models. *Statistica Sinica*. Advance online publication. <https://doi.org/10.5705/ss.202022.0323>
- [10] KAUL, A., FOTOPOULOS, S. B., JANDHYALA, V. K. and SAFIKHANI, A. (2021). Inference on the change point under a high dimensional sparse mean shift. *Electronic Journal of Statistics* **15** 71–134. <https://doi.org/10.1214/20-EJS1791>. MR4195770
- [11] KILLICK, R., FEARNHEAD, P. and ECKLEY, I. A. (2012). Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association* **107** 1590–1598. <https://doi.org/10.1080/01621459.2012.737745>. MR3036418
- [12] KOVÁCS, S., BÜHLMANN, P., LI, H. and MUNK, A. (2022). Seeded binary segmentation: a general methodology for fast and optimal changepoint detection. *Biometrika* **110** 249–256. <https://doi.org/10.1093/biomet/asac052>. MR4565454
- [13] LIU, H., GAO, C. and SAMWORTH, R. J. (2021). Minimax rates in sparse, high-dimensional change point detection. *The Annals of Statistics* **49** 1081–1112. <https://doi.org/10.1214/20-AOS1994>. MR4255120
- [14] MOEN, P. A. J. (2024). *HDCCD: High-Dimensional Changepoint Detection*. CRAN. <https://cran.r-project.org/package=HDCCD>.
- [15] PILLIAT, E., CARPENTIER, A. and VERZELEN, N. (2023). Optimal multiple changepoint detection for high-dimensional data. *Electronic Journal of Statistics* **17** 1240–1315. <https://doi.org/10.1214/23-EJS2126>. MR4576243
- [16] SAMPFORD, M. R. (1953). Some inequalities on Mill’s ratio and related functions. *The Annals of Mathematical Statistics* **24** 130–132. <https://doi.org/10.1214/aoms/1177729093>. MR0054890
- [17] SHAKED, M. and SHANTHIKUMAR, J. G. (2007). *Stochastic Orders*, 1st edition ed. *Springer Series in Statistics*. Springer-Verlag New York. <https://doi.org/10.1007/978-0-387-34675-5>. MR2265633
- [18] TICKLE, S. (2022). *SUBSET*. GitHub. <https://github.com/SOTickle/SUBSET>.
- [19] TICKLE, S. O., ECKLEY, I. A. and FEARNHEAD, P. (2021). A computationally efficient, high-dimensional multiple changepoint procedure with application to global terrorism incidence. *Journal of the Royal Statistical*

- Society: Series A (Statistics in Society)* **184** 1303–1325. <https://doi.org/10.1111/rssa.12695>. MR4344638
- [20] TVETEN, M., ECKLEY, I. A. and FEARNHEAD, P. (2022). Scalable change-point and anomaly detection in cross-correlated data with an application to condition monitoring. *The Annals of Applied Statistics* **16** 721–743. <https://doi.org/10.1214/21-A0AS1508>. MR4438809
- [21] WANG, T. and SAMWORTH, R. J. (2018). High dimensional change point estimation via sparse projection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **80** 57–83. <https://doi.org/10.1111/rssb.12243>. MR3744712
- [22] WANG, D., YU, Y. and RINALDO, A. (2020). Univariate mean change point detection: Penalization, CUSUM and optimality. *Electronic Journal of Statistics* **14** 1917–1961. <https://doi.org/10.1214/20-EJS1710>. MR4091859