ROYAL STATISTICAL SOCIETY
DATA | EVIDENCE | DECISIONS

Journal of the Royal Statistical Society
Statistical Methodology
Series B

J. R. Statist. Soc. B (2018)
80, *Part* 1, *pp.* 57–83

High dimensional change point estimation via sparse projection

Tengyao Wang and Richard J. Samworth
University of Cambridge, UK

[Received June 2016. Final revision June 2017]

Summary. Change points are a very common feature of 'big data' that arrive in the form of a data stream. We study high dimensional time series in which, at certain time points, the mean structure changes in a sparse subset of the co-ordinates. The challenge is to borrow strength across the co-ordinates to detect smaller changes than could be observed in any individual component series. We propose a two-stage procedure called inspect for estimation of the change points: first, we argue that a good projection direction can be obtained as the leading left singular vector of the matrix that solves a convex optimization problem derived from the cumulative sum transformation of the time series. We then apply an existing univariate change point estimation algorithm to the projected series. Our theory provides strong guarantees on both the number of estimated change points and the rates of convergence of their locations, and our numerical studies validate its highly competitive empirical performance for a wide range of data-generating mechanisms. Software implementing the methodology is available in the R package InspectChangepoint.

Keywords: Change point estimation; Convex optimization; Dimension reduction; Piecewise stationary; Segmentation; Sparsity

1. Introduction

One of the most commonly encountered issues with 'big data' is heterogeneity. When collecting vast quantities of data, it is usually unrealistic to expect that stylized, traditional statistical models of independent and identically distributed (IID) observations can adequately capture the complexity of the underlying data-generating mechanism. Departures from such models may take many forms, including missing data, correlated errors and data combined from multiple sources, to mention just a few.

When data are collected over time, heterogeneity often manifests itself through non-stationarity, where the data-generating mechanism varies with time. Perhaps the simplest form of non-stationarity assumes that population changes occur at a relatively small number of discrete time points. If correctly estimated, these 'change points' can be used to partition the original data set into shorter segments, which can then be analysed by using methods designed for stationary time series. Moreover, the locations of these change points are often themselves of significant practical interest.

In this paper, we study high dimensional time series that may have change points; moreover, we consider in particular settings where, at a change point, the mean structure changes in a sparse subset of the co-ordinates. Despite their simplicity, such models are of great interest in a wide variety of applications. For instance, in the case of stock price data, it may be

Address for correspondence: Richard J. Samworth, Statistical Laboratory, Centre for Mathematical Sciences, Wilberforce Road, Cambridge, CB3 0WB, UK.
E-mail: r.samworth@statslab.cam.ac.uk

1369–7412/18/80057

that stocks in related industry sectors experience virtually simultaneous 'shocks' (Chen and Gupta, 1997). In Internet security monitoring, a sudden change in traffic at multiple routers may be an indication of a distributed denial of service attack (Peng *et al.*, 2004). In functional magnetic resonance imaging studies, a rapid change in blood oxygen level dependent contrast in a subset of voxels may suggest neurological activity of interest (Aston and Kirch, 2012).

Our main contribution is to propose a new method for estimating the number and locations of the change points in such high dimensional time series, which is a challenging task in the absence of knowledge of the co-ordinates that undergo a change. In brief, we first seek a good projection direction, which should ideally be closely aligned with the vector of mean changes. We can then apply an existing univariate change point estimation algorithm to the projected series. For this reason, we call our algorithm inspect, short for informative sparse projection for estimation of change points; it is implemented in the R package InspectChangepoint (Wang and Samworth, 2016).

In more detail, in the single-change-point case, our first observation is that, at the population level, the vector of mean changes is the leading left singular vector of the matrix obtained as the cumulative sum (CUSUM) transformation of the mean matrix of the time series. This motivates us to begin by applying the CUSUM transformation to the time series. Unfortunately, computing the k-sparse leading left singular vector of a matrix is a combinatorial optimization problem, but nevertheless we can formulate an appropriate convex relaxation of the problem, from which we derive our projection direction. At the second stage of our algorithm, we compute the vector of CUSUM statistics for the projected series, identifying a change point if the maximum absolute value of this vector is sufficiently large. For the case of multiple change points, we combine our single-change-point algorithm with the method of wild binary segmentation (Fryzlewicz, 2014) to identify change points recursively.

A brief illustration of the inspect algorithm in action is given in Fig. 1. Here, we simulated a 2000×1000 data matrix having independent normal columns with identity covariance and with three change points in the mean structure at locations 500, 1000 and 1500. Changes occur in 40 co-ordinates, where consecutive change points overlap in half of their co-ordinates, and the squared l_2-norms of the vectors of mean changes were 0.4, 0.9 and 1.6 respectively. Fig. 1(a) shows the original data matrix and Fig. 1(b) shows its CUSUM transformation, whereas Fig. 1(c) shows overlays for the three change points detected of the univariate CUSUM statistics after projection. Finally, Fig. 1(d) displays the largest absolute values of the projected CUSUM statistics obtained by running the wild binary segmentation algorithm to completion (in practice, we would apply a termination criterion instead, but this is still helpful for illustration). We see that the three detected change points are very close to their true locations, and it is only for these three locations that we obtain a sufficiently large CUSUM statistic to declare a change point. We emphasize that our focus here is on the so-called *offline* version of the change point estimation problem, where we observe the whole data set before seeking to locate change points. The corresponding on-line problem, where one aims to declare a change point as soon as possible after it has occurred, is also of great interest (Tartakovsky *et al.*, 2014) but is beyond the scope of the current work.

Our theoretical development proceeds first by controlling the angle between the estimated projection direction and the optimal direction, which is given by the normalized vector of mean changes. Under appropriate conditions, this enables us to provide finite sample bounds which guarantee that with high probability we both recover the correct number of change points and estimate their locations to within a specified accuracy. Indeed, in the single-change-point case, the rate of convergence for the change point location estimation of our method is within a doubly

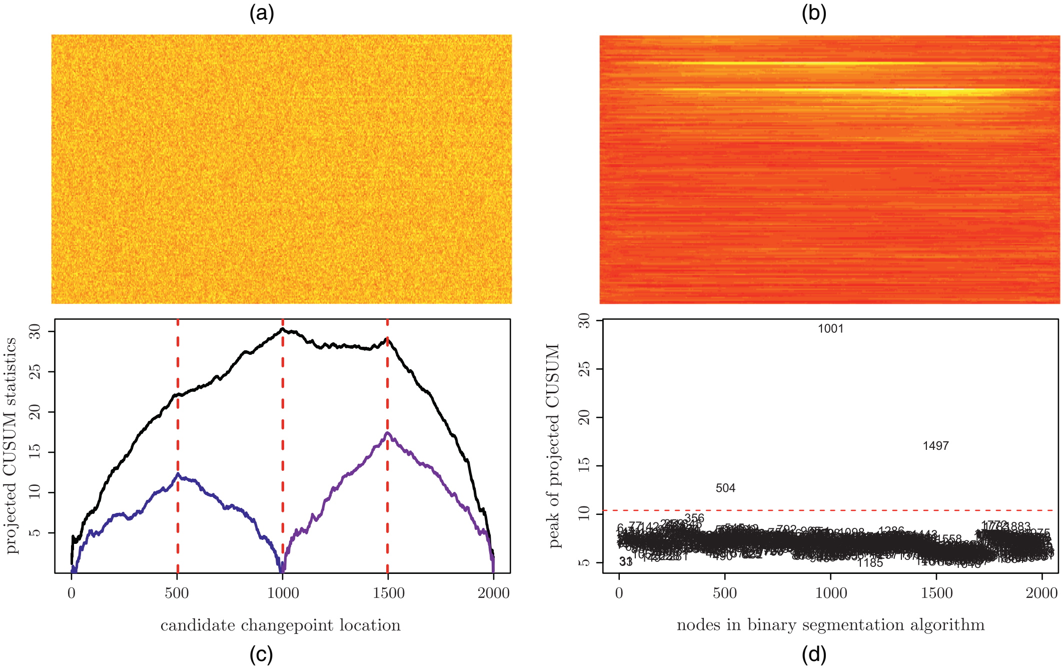


Fig. 1. Example of the inspect algorithm in action: (a) visualization of the data matrix; (b) its CUSUM transformation; (c) overlay of the projected CUSUM statistics for the three change points detected; (d) visualization of thresholding; the three change points detected are above the threshold (------), whereas the remaining numbers are the test statistics obtained if we run wild binary segmentation to completion without applying a termination criterion

logarithmic factor of the minimax optimal rate. Our extensive numerical studies indicate that the algorithm performs extremely well in a wide variety of settings.

The study of change point problems dates at least back to Page (1955) and has since found applications in many areas, including genetics (Olshen *et al.*, 2004), disease outbreak watch (Sparks *et al.*, 2010) and aerospace engineering (Henry *et al.*, 2010), in addition to those already mentioned. There is a vast and rapidly growing literature on different methods for change point detection and localization, especially in the univariate problem. Surveys of various methods can be found in Csörgő and Horváth (1997) and Horváth and Rice (2014). In the case of univariate change point estimation, state of the art methods include the pruned exact linear time method (Killick *et al.*, 2012), wild binary segmentation (Fryzlewicz, 2014) and simultaneous multiscale change point estimator (Frick *et al.*, 2014).

Some of the univariate change point methodologies have been extended to multivariate settings. Examples include Horváth *et al.* (1999), Ombao *et al.* (2005), Aue *et al.* (2009) and Kirch *et al.* (2015). However, there are fewer available tools for high dimensional change point problems, where both the dimension p and the length n of the data stream may be large, and where we may allow a sparsity assumption on the co-ordinates of change. Bai (2010) investigated the performance of the least squares estimator of a single change point in the high dimensional setting. Zhang *et al.* (2010), Horváth and Hušková (2012) and Enikeeva and Harchaoui (2014) considered estimators based on l_2-aggregations of CUSUM statistics in all co-ordinates, but without using any sparsity assumptions. Enikeeva and Harchaoui (2014) also considered a scan statistic that takes sparsity into account. Jirak (2015) considered an l_∞-aggregation of the

CUSUM statistics that works well for sparse change points. Cho and Fryzlewicz (2015) proposed sparse binary segmentation, which also takes sparsity into account and can be viewed as a hard thresholding of the CUSUM matrix followed by an l_1-aggregation. Cho (2016) proposes a double-CUSUM algorithm that performs a CUSUM transformation along the location axis on the columnwise-sorted CUSUM matrix. In a slightly different setting, Lavielle and Teyssiere (2006), Aue *et al.* (2009), Bücher *et al.* (2014), Preuß *et al.* (2015) and Cribben and Yu (2015) dealt with changes in cross-covariance, whereas Soh and Chandrasekaran (2017) studied a high dimensional change point problem where all mean vectors are sparse. Aston and Kirch (2014) considered the asymptotic efficiency of detecting a single change point in a high dimensional setting, and the oracle projection-based estimator under cross-sectional dependence structure.

The outline of the rest of the paper is as follows. In Section 2, we give a formal description of the problem and the class of data-generating mechanisms under which our theoretical results hold. Our methodological development in the single-change-point setting is presented in Section 3 and includes theoretical guarantees on both the projection direction and location of the estimated change point in the simplest case of observations that are independent across both space and time. Section 4 extends these ideas to the case of multiple change points with the aid of wild binary segmentation, and our numerical studies are given in Section 5. Section 6 studies in detail important cases of temporal and spatial dependence. For temporal dependence, no change to our methodology is required, but new arguments are needed to provide theoretical guarantees; for spatial dependence, we show how to modify our methodology to try to maximize the signal-to-noise ratio of the projected univariate series, and we also provide corresponding theoretical results on the performance of this variant of the basic inspect algorithm. Proofs of our main results are given in Appendix A, with the exception of the (lengthy) proof of theorem 2; the proof of this result, together with additional results and their proofs are given in the on-line supplementary material, hereafter referred to simply as the on-line supplement.

We conclude this section by introducing some notation that is used throughout the paper. For a vector $u=(u_1,\ldots,u_M)^{\mathrm{T}}\in\mathbb{R}^M$, a matrix $A=(A_{ij})\in\mathbb{R}^{M\times N}$ and for $q\in[1,\infty)$, we write $\|u\|_q := (\Sigma_{i=1}^M|u_i|^q)^{1/q}$ and $\|A\|_q := (\Sigma_{i=1}^M\Sigma_{j=1}^N|A_{ij}|^q)^{1/q}$ for their (entrywise) l_q-norms, as well as $\|u\|_\infty := \max_{i=1,\ldots,M}|u_i|$ and $\|A\|_\infty := \max_{i=1,\ldots,M,j=1,\ldots,N}|A_{ij}|$. We write $\|A\|_* := \Sigma_{i=1}^{\min(M,N)}\sigma_i(A)$ and $\|A\|_{\mathrm{op}} := \max_i\sigma_i(A)$ respectively for the nuclear norm and operator norm of matrix A, where $\sigma_1(A),\ldots,\sigma_{\min(M,N)}(A)$ are its singular values. We also write $\|u\|_0 := \Sigma_{i=1}^M\mathbb{1}_{\{u_i\neq 0\}}$. For $S\subseteq\{1,\ldots,M\}$ and $T\subseteq\{1,\ldots,N\}$, we write $u_S := (u_i : i\in S)^{\mathrm{T}}$ and write $M_{S,T}$ for the $|S|\times|T|$ submatrix of A obtained by extracting the rows and columns with indices in S and T respectively. For two matrices $A, B\in\mathbb{R}^{M\times N}$, we denote their trace inner product as $\langle A,B\rangle = \mathrm{tr}(A^{\mathrm{T}}B)$. For two non-zero vectors $u, v\in\mathbb{R}^p$, we write

$$\angle(u,v) := \cos^{-1}\left(\frac{|\langle u,v\rangle|}{\|u\|_2\|v\|_2}\right)$$

for the acute angle bounded between them. We let $\mathbb{S}^{p-1} := \{x\in\mathbb{R}^p : \|x\|_2 = 1\}$ be the unit Euclidean sphere in $\mathbb{R}^p$, and let $\mathbb{S}^{p-1}(k) := \{x\in\mathbb{S}^{p-1} : \|x\|_0 \leqslant k\}$. Finally, we write $a_n \asymp b_n$ to mean $0<\liminf_{n\to\infty}|a_n/b_n| \leqslant \limsup_{n\to\infty}|a_n/b_n| < \infty$.

2. Problem description

We initially study the following basic independent time series model: let $X_1,\ldots,X_n$ be independent p-dimensional random vectors sampled from

$$X_t \sim N_p(\mu_t, \sigma^2 I_p), \qquad 1\leqslant t\leqslant n, \tag{1}$$

and combine the observations into a matrix $X=(X_1,\ldots,X_n)\in\mathbb{R}^{p\times n}$. Extensions to settings of both temporal and spatial dependence will be studied in detail in Section 6. We assume that the mean vectors follow a piecewise constant structure with $\nu+1$ segments. In other words, there are ν *change points*

$$1\leqslant z_1<z_2<\ldots<z_\nu\leqslant n-1$$

such that

$$\mu_{z_i+1}=\ldots=\mu_{z_{i+1}}=:\mu^{(i)}, \qquad \forall\, 0\leqslant i\leqslant \nu, \tag{2}$$

where we adopt the convention that $z_0:=0$ and $z_{\nu+1}:=n$. For $i=1,\ldots,\nu$, write

$$\theta^{(i)}:=\mu^{(i)}-\mu^{(i-1)} \tag{3}$$

for the (non-zero) difference in means between consecutive stationary segments. We shall later assume that the changes in mean are sparse in the sense that there exists $k\in\{1,\ldots,p\}$ (typically k is much smaller than p) such that

$$\|\theta^{(i)}\|_0\leqslant k \tag{4}$$

for each $i=1,\ldots,\nu$, since our methodology performs best when aggregating signals spread across an (unknown) sparse subset of co-ordinates; see also the discussion after corollary 2 below. However, we remark that our methodology does not require knowledge of the level of sparsity and can be applied in non-sparse settings as well.

Our goal is to estimate the set of change points $\{z_1,\ldots,z_\nu\}$ in the high dimensional regime, where p may be comparable with, or even larger than, the length n of the series. The signal strength of the estimation problem is determined by the magnitude of mean changes $\{\theta^{(i)}:1\leqslant i\leqslant\nu\}$ and the lengths of stationary segments $\{z_{i+1}-z_i:0\leqslant i\leqslant\nu\}$, whereas the noise is related to the variance σ^2 and the dimensionality p of the observed data points. For our theoretical results, we shall assume that the change point locations satisfy

$$n^{-1}\min\{z_{i+1}-z_i:0\leqslant i\leqslant\nu\}\geqslant\tau, \tag{5}$$

and the magnitudes of mean changes are such that

$$\|\theta^{(i)}\|_2\geqslant\vartheta, \qquad \forall\, 1\leqslant i\leqslant\nu. \tag{6}$$

Suppose that an estimation procedure outputs $\hat{\nu}$ change points at $1\leqslant\hat{z}_1<\ldots<\hat{z}_{\hat{\nu}}\leqslant n-1$. Our finite sample bounds will imply a rate of convergence for inspect in an asymptotic setting where the problem parameters are allowed to depend on n. Suppose that $\mathcal{P}_n$ is a class of distributions of $X\in\mathbb{R}^{p\times n}$ with sample size n. In this context, we follow the convention in the literature (e.g. Venkatraman (1992)) and say that the procedure is consistent for $\mathcal{P}_n$ with rate of convergence ρ_n if

$$\inf_{P\in\mathcal{P}_n}\mathbb{P}_P(\hat{\nu}=\nu \text{ and } |\hat{z}_i-z_i|\leqslant n\rho_n \text{ for all } 1\leqslant i\leqslant\nu)\rightarrow 1 \tag{7}$$

as $n\rightarrow\infty$.

3. Data-driven projection estimator for a single change point

We first consider the problem of estimating a single change point (i.e. $\nu=1$) in a high dimensional data set $X\in\mathbb{R}^{p\times n}$. Our initial focus will be on the independent time series setting that was outlined in Section 2, but our analysis in Section 6 will show how these ideas can be generalized to cases of temporal dependence. For simplicity, write $z:=z_1$, $\theta=(\theta_1,\ldots,\theta_p)^{\mathrm{T}}:=\theta^{(1)}$ and $\tau:=n^{-1}\min\{z,n-z\}$. We seek to aggregate the rows of the data matrix X in an almost optimal way to maximize the signal-to-noise ratio, and then to locate the change point by using a one-dimensional procedure. For any $a \in \mathbb{S}^{p-1}$, $a^{\mathrm{T}} X$ is a one-dimensional time series with

$$a^{\mathrm{T}} X_t \sim N(a^{\mathrm{T}} \mu_t, \sigma^2).$$

Hence, the choice $a = \theta/\|\theta\|_2$ maximizes the magnitude of the difference in means between the two segments. However, θ is typically unknown in practice, so we should seek a projection direction that is close to the oracle projection direction $v := \theta/\|\theta\|_2$. Our strategy is to perform sparse singular value decomposition on the CUSUM transformation of X. The method and limit theory of CUSUM statistics in the univariate case can be traced back to Darling and Erdős (1956). For $p \in \mathbb{N}$ and $n \geqslant 2$, we define the CUSUM transformation $\mathcal{T}_{p,n} : \mathbb{R}^{p\times n} \to \mathbb{R}^{p\times(n-1)}$ by

$$\begin{aligned}[\mathcal{T}_{p,n}(M)]_{j,t} &:= \sqrt{\left\{\frac{t(n-t)}{n}\right\}}\left(\frac{1}{n-t}\sum_{r=t+1}^{n} M_{j,r} - \frac{1}{t}\sum_{r=1}^{t} M_{j,r}\right)\\ &= \sqrt{\left\{\frac{n}{t(n-t)}\right\}}\left(\frac{t}{n}\sum_{r=1}^{n} M_{j,r} - \sum_{r=1}^{t} M_{j,r}\right). \qquad (8)\end{aligned}$$

In fact, to simplify the notation, we shall write $\mathcal{T}$ for $\mathcal{T}_{p,n}$, since p and n can be inferred from the dimensions of the argument of $\mathcal{T}$. Note also that $\mathcal{T}$ reduces to computing the vector of classical one-dimensional CUSUM statistics when $p = 1$. We write

$$X = \boldsymbol{\mu} + W,$$

where $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_n) \in \mathbb{R}^{p\times n}$ and $W = (W_1, \ldots, W_n)$ is a $p \times n$ random matrix with independent $N_p(0, \sigma^2 I_p)$ columns. Let $T := \mathcal{T}(X)$, $A := \mathcal{T}(\boldsymbol{\mu})$ and $E := \mathcal{T}(W)$, so by the linearity of the CUSUM transformation we have the decomposition

$$T = A + E.$$

We remark that, when σ is known, each $|T_{j,t}|$ is the likelihood ratio statistic for testing the null hypothesis that the jth row of $\boldsymbol{\mu}$ is constant against the alternative that the jth row of $\boldsymbol{\mu}$ undergoes a single change at time t. Moreover, if the direction $v \in \mathbb{S}^{p-1}$ of the potential single change at a given time t were known, then the most powerful test of whether or not $\vartheta = 0$ would be based on $|(v^{\mathrm{T}} T)_t|$. In the single-change-point case, the entries of the matrix A can be computed explicitly:

$$A_{j,t} = \begin{cases} \sqrt{\left\{\dfrac{t}{n(n-t)}\right\}}(n-z)\theta_j, & \text{if } t \leqslant z,\\[2ex] \sqrt{\left(\dfrac{n-t}{nt}\right)} z\theta_j, & \text{if } t > z.\end{cases}$$

Hence we can write

$$A = \theta\gamma^{\mathrm{T}}, \qquad (9)$$

where

$$\gamma := \frac{1}{\sqrt{n}}\left(\sqrt{\left(\frac{1}{n-1}\right)}(n-z), \sqrt{\left(\frac{2}{n-2}\right)}(n-z), \ldots, \sqrt{\{z(n-z)\}}, \sqrt{\left(\frac{n-z-1}{z+1}\right)}z, \ldots, \sqrt{\left(\frac{1}{n-1}\right)}z\right)^{\mathrm{T}}. \qquad (10)$$

In particular, this implies that the oracle projection direction is the leading left singular vector of the rank 1 matrix A. In the ideal case where k is known, we could in principle let $\hat{v}_{\max,k}$ be a k-sparse leading left singular vector of T, defined by

$$\hat{v}_{\max,k} \in \arg\max_{\tilde{v} \in \mathbb{S}^{p-1}(k)} \|T^{\mathrm{T}} \tilde{v}\|_2, \tag{11}$$

and it can then be shown by using a perturbation argument akin to the Davis–Kahan ‘$\sin(\theta)$’ theorem (see Davis and Kahan (1970) and Yu *et al.* (2015)) that $\hat{v}_{\max,k}$ is a consistent estimator of the oracle projection direction v under mild conditions (see proposition 1 in the on-line supplement). However, the optimization problem (11) is non-convex and hard to implement. In fact, computing the k-sparse leading left singular vector of a matrix is known to be ‘NP hard’ (e.g. Tillmann and Pfetsch (2014)). The naive algorithm that scans through all possible k-subsets of the rows of T has running time exponential in k, which quickly becomes impractical to run for even moderate sizes of k.

A natural approach to remedy this computational issue is to work with a convex relaxation of the optimization problem (11) instead. In fact, we can write

$$\begin{aligned} \max_{u \in \mathbb{S}^{p-1}(k)} \|u^{\mathrm{T}} T\|_2 &= \max_{u \in \mathbb{S}^{p-1}(k), w \in \mathbb{S}^{n-2}} u^{\mathrm{T}} T w \\ &= \max_{u \in \mathbb{S}^{p-1}, w \in \mathbb{S}^{n-2}, \|u\|_0 \leqslant k} \langle u w^{\mathrm{T}}, T \rangle = \max_{M \in \mathcal{M}} \langle M, T \rangle, \end{aligned} \tag{12}$$

where $\mathcal{M} := \{M \in \mathbb{R}^{p \times (n-1)} : \|M\|_* = 1, \mathrm{rank}(M) = 1, M \text{ has at most } k \text{ non-zero rows}\}$. The final expression in equation (12) has a convex (linear) objective function $M \mapsto \langle M, T \rangle$. The requirement $\mathrm{rank}(M) = 1$ in the constraint set $\mathcal{M}$ is equivalent to $\|\sigma(M)\|_0 = 1$, where $\sigma(M) := (\sigma_1(M), \ldots, \sigma_{\min(p,n-1)}(M))^{\mathrm{T}}$ is the vector of singular values of M. This motivates us to absorb the rank constraint into the nuclear norm constraint, which we relax from an equality constraint to an inequality constraint to make it convex. Furthermore, we can relax the row sparsity constraint in the definition of $\mathcal{M}$ to an entrywise l_1-norm penalty. The optimization problem of finding

$$\hat{M} \in \arg\max_{M \in \mathcal{S}_1} \{\langle T, M \rangle - \lambda \|M\|_1\}, \tag{13}$$

where $\mathcal{S}_1 := \{M \in \mathbb{R}^{p \times (n-1)} : \|M\|_* \leqslant 1\}$ and $\lambda > 0$ is a tuning parameter to be chosen later, is therefore a convex relaxation of problem (11). We remark that a similar convex relaxation has appeared in the different context of sparse principal component estimation (d’Aspremont *et al.*, 2007), where the sparse leading left singular vector is also the optimization target. The convex problem (13) may be solved using the alternating direction method of multipliers algorithm (see Gabay and Mercier (1976) and Boyd *et al.* (2011)) as in algorithm 1 (Table 1). More specifically, the optimization problem (13) is equivalent to maximizing $\langle T, Y \rangle - \lambda \|Z\|_1 - \mathbb{I}_{\mathcal{S}_1}(Y)$ subject to $Y = Z$, where $\mathbb{I}_{\mathcal{S}_1}$ is the function that is 0 on $\mathcal{S}_1$ and ∞ on $\mathcal{S}_1^{\mathrm{c}}$. Its augmented Lagrangian is given by

$$L(Y, Z, R) := \langle T, Y \rangle - \mathbb{I}_{\mathcal{S}_1}(Y) - \lambda \|Z\|_1 - \langle R, Y - Z \rangle - \tfrac{1}{2} \|Y - Z\|_2^2,$$

with the Lagrange multiplier R being the dual variable. Each iteration of the main loop in algorithm 1 first performs a primal update by maximizing $L(Y, Z, R)$ marginally with respect to Y and Z, then followed by a dual gradient update of R with constant step size. The function $\Pi_{\mathcal{S}_1}(\cdot)$ in algorithm 1 denotes projection onto the convex set $\mathcal{S}_1$ with respect to the Frobenius norm distance. If $A = UDV^{\mathrm{T}}$ is the singular value decomposition of $A \in \mathbb{R}^{p \times (n-1)}$ with $\mathrm{rank}(A) = r$, where D is a diagonal matrix with diagonal entries $d_1, \ldots, d_r$, then $\Pi_{\mathcal{S}_1}(A) = U\tilde{D}V^{\mathrm{T}}$, where $\tilde{D}$ is a diagonal matrix with entries $\tilde{d}_1, \ldots, \tilde{d}_r$ such that $(\tilde{d}_1, \ldots, \tilde{d}_r)^{\mathrm{T}}$ is the Euclidean projection of

Table 1. Algorithm 1: pseudocode for an alternating direction method of multipliers algorithm that computes the solution to the optimization problem (13)

```
Input: $T \in \mathbb{R}^{p\times(n-1)}$, $\lambda > 0$
Set: $Y = Z = R = \mathbf{0} \in \mathbb{R}^{p\times(n-1)}$
repeat
  $Y \leftarrow \Pi_{\mathcal{S}_1}(Z - R + T)$
  $Z \leftarrow \mathrm{soft}(Y + R, \lambda)$
  $R \leftarrow R + (Y - Z)$
until $Y - Z$ converges to 0
$\hat{M} \leftarrow Y$
Output: $\hat{M}$
```

the vector $(d_1,\ldots,d_r)^{\mathrm{T}}$ onto the standard $(r-1)$-simplex

$$\Delta^{r-1} := \left\{ (x_1,\ldots,x_r)^{\mathrm{T}} \in \mathbb{R}^r : \sum_{l=1}^{r} x_l = 1 \text{ and } x_l \geqslant 0 \text{ for all } l \right\}.$$

For an efficient algorithm for such simplicial projection, see Chen and Ye (2011). The soft function in algorithm 1 denotes an entrywise soft thresholding operator defined by $(\mathrm{soft}(A,\lambda))_{ij} := \mathrm{sgn}(A_{ij}) \max\{|A_{ij}| - \lambda, 0\}$ for any $\lambda \geqslant 0$ and matrix $A = (A_{ij})$.

We remark that one may be interested to relax problem (13) further by replacing $\mathcal{S}_1$ with the larger set $\mathcal{S}_2 := \{M \in \mathbb{R}^{p\times(n-1)} : \|M\|_2 \leqslant 1\}$ defined by the entrywise l_2-unit ball. We see from proposition 2 in the on-line supplement that the smoothness of $\mathcal{S}_2$ results in a simple dual formulation, which implies that

$$\tilde{M} := \frac{\mathrm{soft}(T,\lambda)}{\|\mathrm{soft}(T,\lambda)\|_2} = \arg\max_{M\in\mathcal{S}_2} \{\langle T, M\rangle - \lambda\|M\|_1\} \tag{14}$$

is the unique optimizer of the primal problem. The soft thresholding operation is significantly faster than the alternating direction method of multipliers algorithm in algorithm 1. Hence by enlarging $\mathcal{S}_1$ to $\mathcal{S}_2$, we can significantly speed up the running time of the algorithm in exchange for some loss in statistical efficiency caused by the further relaxation of the constraint set. See Section 5 for further discussion.

Let $\hat{v}$ be the leading left singular vector of

$$\hat{M} \in \arg\max_{M\in\mathcal{S}} \{\langle T, M\rangle - \lambda\|M\|_1\}, \tag{15}$$

for either $\mathcal{S} = \mathcal{S}_1$ or $\mathcal{S} = \mathcal{S}_2$. To describe the theoretical properties of $\hat{v}$ as an estimator of the oracle projection direction v, we introduce the following class of distributions: let $\mathcal{P}(n, p, k, \nu, \vartheta, \tau, \sigma^2)$ denote the class of distributions of $X = (X_1,\ldots,X_n) \in \mathbb{R}^{p\times n}$ with independent columns drawn from distribution (1), where the change point locations satisfy condition (5) and the vectors of mean changes are such that conditions (4) and (6) hold. Although this notation accommodates the multiple-change-point setting that is studied in Section 4 below, we emphasize that our focus here is on the single-change-point setting. The error bound in proposition 1 below relies on a generalization of the curvature lemma in Vu *et al.* (2013), lemma 3.1, presented as lemma 6 in the on-line supplement.

Proposition 1. Suppose that $\hat{M}$ satisfies expression (15) for either $\mathcal{S}=\mathcal{S}_1$ or $\mathcal{S}=\mathcal{S}_2$. Let $\hat{v}$ be the leading left singular vector of $\hat{M}$. If $n \geqslant 6$ and if we choose $\lambda \geqslant 2\sigma\sqrt{\log\{p\log(n)\}}$, then

$$\sup_{P\in\mathcal{P}(n,p,k,1,\vartheta,\tau,\sigma^2)} \mathbb{P}_P\left\{\sin\angle(\hat{v},v) > \frac{32\lambda\sqrt{k}}{\tau\vartheta\sqrt{n}}\right\} \leqslant \frac{4}{\{p\log(n)\}^{1/2}}.$$

The following corollary restates the rate of convergence of the projection estimator in a simple asymptotic regime.

Corollary 1. Consider an asymptotic regime where $\log(p)=O\{\log(n)\}$, σ is a constant, $\vartheta \asymp n^{-a}$, $\tau \asymp n^{-b}$ and $k \asymp n^c$ for some $a\in\mathbb{R}$, $b\in[0,1]$ and $c\geqslant 0$. Then, setting $\lambda := 2\sigma\sqrt{\log\{p\log(n)\}}$ and provided that $a+b+c/2<\frac{1}{2}$, we have for every $\delta>0$ that

$$\sup_{P\in\mathcal{P}(n,p,k,1,\vartheta,\tau,\sigma^2)} \mathbb{P}_P\{\angle(\hat{v},v) > n^{-(1-2a-2b-c)/2+\delta}\} \to 0.$$

Proposition 1 and corollary 1 illustrate the benefits of assuming that the changes in mean structure occur only in a sparse subset of the co-ordinates. Indeed, these results mimic similar findings in other high dimensional statistical problems where sparsity plays a key role, indicating that one pays a logarithmic price for absence of knowledge of the true sparsity set. See, for instance, Bickel *et al.* (2009) in the context of the lasso in high dimensional linear models, or Johnstone and Lu (2009), or Wang *et al.* (2016) in the context of sparse principal component analysis.

After obtaining a good estimator $\hat{v}$ of the oracle projection direction, the natural next step is to project the data matrix X along the direction $\hat{v}$, and to apply an existing one-dimensional change point localization method on the projected data. In this work, we apply a one-dimensional CUSUM transformation to the projected series and estimate the change point by the location of the maximum of the CUSUM vector. Our overall procedure for locating a single change point in a high dimensional time series is given in algorithm 2 (Table 2). In our description of this algorithm, the noise level σ is assumed to be known. If σ is unknown, we can estimate it robustly using, for example, the median absolute deviation of the marginal one-dimensional series (Hampel, 1974). For convenience of later reference, we have required algorithm 2 to output both the estimated change point location $\hat{z}$ and the associated maximum absolute post-projection one-dimensional CUSUM statistic $\bar{T}_{\max}$.

From a theoretical point of view, the fact that $\hat{v}$ is estimated by using the entire data set X makes it difficult to analyse the post-projection noise structure. For this reason, in the analysis below, we work with a slight variant of algorithm 2. We assume for convenience that $n=2n_1$ is even, and define $X^{(1)}, X^{(2)} \in \mathbb{R}^{p\times n_1}$ by

Table 2. Algorithm 2: pseudocode for a single high dimensional change point estimation algorithm

```
*Input*: $X\in\mathbb{R}^{p\times n}$, $\lambda>0$
*Step 1*: perform the CUSUM transformation $T \leftarrow \mathcal{T}(X)$
*Step 2*: use algorithm 1 or equation (14) (with inputs $T$ and $\lambda$ in either case) to solve for an optimizer $\hat{M}$ of expression (15) for $\mathcal{S}=\mathcal{S}_1$ or $\mathcal{S}=\mathcal{S}_2$
*Step 3*: find $\hat{v}\in\arg\max_{\tilde{v}\in\mathbb{S}^{p-1}} \|\hat{M}^{\mathrm{T}}\tilde{v}\|_2$
*Step 4*: let $\hat{z}\in\arg\max_{1\leqslant t\leqslant n-1} |\hat{v}^{\mathrm{T}}T_t|$, where $T_t$ is the $t$th column of $T$, and set $\bar{T}_{\max} \leftarrow |\hat{v}^{\mathrm{T}}T_{\hat{z}}|$
*Output*: $\hat{z}$, $\bar{T}_{\max}$
```

Table 3. Algorithm 3: pseudocode for a sample splitting variant of algorithm 2

```
Input: $X \in \mathbb{R}^{p\times n}$, $\lambda > 0$
Step 1: perform the CUSUM transformation $T^{(1)} \leftarrow \mathcal{T}(X^{(1)})$ and $T^{(2)} \leftarrow \mathcal{T}(X^{(2)})$
Step 2: use algorithm 1 or equation (14) (with inputs $T^{(1)}$, $\lambda$ in either case) to solve for $\hat{M}^{(1)} \in \arg\max_{M\in\mathcal{S}}\{\langle T^{(1)}, M\rangle - \lambda\|M\|_1\}$ with $\mathcal{S} = \{M \in \mathbb{R}^{p\times(n_1-1)} : \|M\|_* \leqslant 1\}$ or $\mathcal{S} = \{M \in \mathbb{R}^{p\times(n_1-1)} : \|M\|_2 \leqslant 1\}$
Step 3: find $\hat{v}^{(1)} \in \arg\max_{\tilde{v}\in\mathbb{S}^{p-1}} \|(\hat{M}^{(1)})^{\mathrm{T}}\tilde{v}\|_2$
Step 4: let $\hat{z} \in 2\arg\max_{1\leqslant t\leqslant n_1-1} |(\hat{v}^{(1)})^{\mathrm{T}}T_t^{(2)}|$, where $T_t^{(2)}$ is the $t$th column of $T^{(2)}$, and set $\bar{T}_{\max} \leftarrow |(\hat{v}^{(1)})^{\mathrm{T}}T_{\hat{z}/2}^{(2)}|$
Output: $\hat{z}, \bar{T}_{\max}$
```

$$X_{j,t}^{(1)} := X_{j,2t-1} \quad \text{and} \quad X_{j,t}^{(2)} := X_{j,2t} \qquad \text{for } 1 \leqslant j \leqslant p,\ 1 \leqslant t \leqslant n_1. \tag{16}$$

We then use $X^{(1)}$ to estimate the oracle projection direction and use $X^{(2)}$ to estimate the change point location after projection (see algorithm 3 (Table 3)). However, we recommend using algorithm 2 in practice to exploit the full signal strength in the data.

We summarize the overall estimation performance of algorithm 3 in the following theorem.

Theorem 1. Suppose that $\sigma > 0$ is known. Let $\hat{z}$ be the output of algorithm 3 with input $X \sim P \in \mathcal{P}(n, p, k, 1, \vartheta, \tau, \sigma^2)$ and $\lambda := 2\sigma\surd\log\{p\log(n)\}$. There exist universal constants $C, C' > 0$ such that, if $n \geqslant 12$ is even, z is even and

$$\frac{C\sigma}{\vartheta\tau}\sqrt{\left[\frac{k\log\{p\log(n)\}}{n}\right]} \leqslant 1, \tag{17}$$

then

$$\mathbb{P}_P\left[\frac{1}{n}|\hat{z} - z| \leqslant \frac{C'\sigma^2\log\{\log(n)\}}{n\vartheta^2}\right] \geqslant 1 - \frac{4}{\{p\log(n/2)\}^{1/2}} - \frac{17}{\log(n/2)}.$$

We remark that, under the conditions of theorem 1, the rate of convergence obtained is minimax optimal up to a factor of $\log\{\log(n)\}$; see proposition 3 in the on-line supplement. It is interesting to note that, once condition (17) is satisfied, the final rate of change point estimation does not depend on τ.

Corollary 2. Suppose that σ is a constant, $\log(p) = O\{\log(n)\}$, $\vartheta \asymp n^{-a}$, $\tau \asymp n^{-b}$ and $k \asymp n^c$ for some $a \in \mathbb{R}$ and $b \in [0, 1]$ and $c \geqslant 0$. If $a + b + c/2 < \frac{1}{2}$, then the output $\hat{z}$ of algorithm 3 with $\lambda := 2\sigma\surd\log\{p\log(n)\}$ is a consistent estimator of the true change point z with rate of convergence $\rho_n = o(n^{-1+2a+\delta})$ for any $\delta > 0$.

Finally in this section, we remark that this asymptotic rate of convergence has previously been observed in Csörgő and Horváth (1997), theorem 2.8.2, for a CUSUM procedure in the special case of univariate observations with τ bounded away from zero (i.e. $b = 0$ in corollary 2 above).

4. Estimating multiple change points

Our algorithm for estimating a single change point can be combined with the wild binary segmentation scheme of Fryzlewicz (2014) to locate sequentially multiple change points in high dimensional time series. The principal idea behind a wild binary segmentation procedure is as follows. We first randomly sample a large number of pairs, $(s_1, e_1), \ldots, (s_Q, e_Q)$ uniformly from the set $\{(l,r)\in\mathbb{Z}^2:0\leqslant l<r\leqslant n\}$, and then apply our single-change-point algorithm to $X^{[q]}$, for $1\leqslant q\leqslant Q$, where $X^{[q]}$ is defined to be the submatrix of X obtained by extracting columns $\{s_q+1,\ldots,e_q\}$ of X. For each $1\leqslant q\leqslant Q$, the single-change-point algorithm (algorithm 2 or 3) will estimate an optimal sparse projection direction $\hat{v}^{[q]}$, compute a candidate change point location $s_q+\hat{z}^{[q]}$ within the time window $[s_q+1,e_q]$ and return a maximum absolute CUSUM statistic $\bar{T}^{[q]}_{\max}$ along the projection direction. We aggregate the Q candidate change point locations by choosing one that maximizes the largest projected CUSUM statistic, $T^{[q]}_{\max}$, as our best candidate. If $T^{[q]}_{\max}$ is above a certain threshold value ξ, we admit the best candidate to the set $\hat{Z}$ of estimated change point locations and repeat the above procedure recursively on the subsegments to the left and right of the estimated change point. Note that, while recursing on a subsegment, we consider only those time windows that are completely contained in the subsegment. The precise algorithm is detailed in algorithm 4 (Table 4).

Algorithm 4 requires three tuning parameters: a regularization parameter λ, a Monte Carlo parameter Q for the number of random time windows and a thresholding parameter ξ that determines termination of recursive segmentation. Theorem 2 below provides choices for λ, Q and ξ that yield theoretical guarantees for consistent estimation of all change points as defined in expression (7).

We remark that if we apply algorithm 2 or 3 on the entire data set X instead of random time windows of X, and then iterate after segmentation, we arrive at a multiple-change-point algorithm based on the classical binary segmentation scheme. The main disadvantage of this classical binary segmentation procedure is its sensitivity to model misspecification. Algorithms 2 and 3 are designed to optimize the detection of a single change point. When we apply them in conjunction with classical binary segmentation to a time series containing more than one change point, the signals from multiple change points may cancel each other out in two different ways that will lead to a loss of power. First, as Fryzlewicz (2014) pointed out in the one-dimensional setting, multiple change points may offset each other in CUSUM computation, resulting in a smaller peak of the CUSUM statistic that is more easily contaminated by

Table 4. Algorithm 4: pseudocode for the multiple-change-point algorithm based on sparse singular vector projection and wild binary segmentation

```
Input: $X\in\mathbb{R}^{p\times n}$, $\lambda>0$, $\xi>0$, $\beta>0$, $Q\in\mathbb{N}$
Step 1: set $\hat{Z}\leftarrow\emptyset$: draw $Q$ pairs of integers $(s_1,e_1),\ldots,(s_Q,e_Q)$ uniformly
    at random from the set $\{(l,r)\in\mathbb{Z}^2:0\leqslant l<r\leqslant n\}$
Step 2: run wbs(0, $n$) where wbs is defined below
Step 3: let $\hat{\nu}\leftarrow|\hat{Z}|$ and sort elements of $\hat{Z}$ in increasing order to yield
    $\hat{z}_1<\ldots<\hat{z}_{\hat{\nu}}$
Output: $\hat{z}_1,\ldots,\hat{z}_{\hat{\nu}}$
Function wbs($s$, $e$)
    Set $\mathcal{Q}_{s,e}\leftarrow\{q:s+n\beta\leqslant s_q<e_q\leqslant e-n\beta\}$
    for $q\in\mathcal{Q}_{s,e}$ do
        run algorithm 2 with $X^{[q]}$, $\lambda$ as input, and let $\hat{z}^{[q]}$, $\bar{T}^{[q]}_{\max}$ be the output
    end
    Find $q_0\in\arg\max_{q\in\mathcal{Q}_{s,e}}\bar{T}^{[q]}_{\max}$ and set $b\leftarrow s_{q_0}+\hat{z}^{[q_0]}$
    if $\bar{T}^{[q_0]}_{\max}>\xi$ then
        $\hat{Z}\leftarrow\hat{Z}\cup\{b\}$
        wbs($s,b$)
        wbs($b,e$)
    end
end
```

the noise. Moreover, in a high dimensional setting, different change points can undergo changes in different sets of (sparse) co-ordinates. This also attenuates the signal strength in the sense that the estimated oracle projection direction from algorithm 1 is aligned to some linear combination of $\theta^{(1)},\ldots,\theta^{(\nu)}$, but not necessarily well aligned to any one particular $\theta^{(i)}$. The wild binary segmentation scheme addresses the model misspecification issue by examining subintervals of the entire time length. When the number of time windows Q is sufficiently large and τ is not too small, with high probability we have reasonably long time windows that contain each individual change point. Hence the single-change-point algorithm will perform well on these segments.

Just as in the case of single-change-point detection, it is easier to analyse the theoretical performance of a sample splitting version of algorithm 4. However, to avoid notational clutter, we shall prove a theoretical result without sample splitting, but with the assumption that, whenever algorithm 2 is used within algorithm 4, its second and third steps (i.e. the steps for estimating the oracle projection direction) are carried out on an independent copy X' of X. We refer to such a variant of the algorithm with an access to an independent sample X' as algorithm 4′. Theorem 2 below, which proves theoretical guarantees of algorithm 4′, can then be readily adapted to work for a sample splitting version of algorithm 4, where we replace n by $n/2$ where necessary.

Theorem 2. Suppose that $\sigma > 0$ is known and $X, X' \overset{\mathrm{IID}}{\sim} P \in \mathcal{P}(n,p,k,\nu,\vartheta,\tau,\sigma^2)$. Let $\hat{z}_1 < \ldots < \hat{z}_{\hat{\nu}}$ be the output of algorithm 4′ with input X, X', $\lambda := 4\sigma\sqrt{\log(np)}$, $\xi := \lambda$, β and Q. Define $\rho = \rho_n := \lambda^2 n^{-1}\vartheta^{-2}\tau^{-4}$, and assume that $n\tau \geqslant 14$. There are universal constants $C, C' > 0$ such that, if $C'\rho < \beta/2 \leqslant \tau/C$ and $C\rho k\tau^2 \leqslant 1$, then

$$\mathbb{P}_P(\hat{\nu} = \nu \text{ and } |\hat{z}_i - z_i| \leqslant C' n\rho \text{ for all } 1 \leqslant i \leqslant \nu) \geqslant 1 - \tau^{-1}\exp(-\tau^2 Q/9) - 6n^{-1}p^{-4}\log(n).$$

Corollary 3. Suppose that σ is a constant, $\vartheta \asymp n^{-a}$, $\tau \asymp n^{-b}$, $k \asymp n^c$ and $\log(p) = O\{\log(n)\}$. If $a + b + c/2 < \frac{1}{2}$ and $2a + 5b < 1$, then there exists $\beta = \beta_n$ such that algorithm 4′ with $\lambda := 4\sigma\sqrt{\log(np)}$ consistently estimates all change points with rate of convergence $\rho_n = o(n^{-(1-2a-4b)+\delta})$ for any $\delta > 0$.

We remark that the consistency that is described in corollary 3 is a rather strong notion, in the sense that it implies convergence in several other natural metrics. For example, if we let

$$d_{\mathrm{H}}(A,B) := \max\{\sup_{a\in A}\inf_{b\in B}|a-b|, \sup_{b\in B}\inf_{a\in A}|a-b|\}$$

denote the Hausdorff distance between non-empty sets A and B on $\mathbb{R}$, then result (7) implies that, with probability tending to 1,

$$\frac{1}{n} d_{\mathrm{H}}(\{\hat{z}_i : 1 \leqslant i \leqslant \hat{\nu}\}, \{z_i : 1 \leqslant i \leqslant \nu\}) \leqslant \rho_n.$$

Similarly, denote the L_1 Wasserstein distance between probability measures P and Q on $\mathbb{R}$ by

$$d_{\mathrm{W}}(P,Q) := \inf_{(U,V)\sim(P,Q)} \mathbb{E}|U - V|,$$

where the infimum is taken over all pairs of random variables U and V defined on the same probability space with $U \sim P$ and $V \sim Q$. Then result (7) also implies that, with probability tending to 1,

$$\frac{1}{n} d_{\mathrm{W}}\left(\frac{1}{\hat{\nu}}\sum_{i=1}^{\hat{\nu}} \delta_{\hat{z}_i}, \frac{1}{\nu}\sum_{i=1}^{\nu} \delta_{z_i}\right) \leqslant \rho_n,$$

where δ_a denotes a Dirac point mass at a.

5. Numerical studies

In this section, we examine the empirical performance of the inspect algorithm in a range of settings and compare it with a variety of other recently proposed methods. In both single- and multiple-change-point scenarios, the implementation of inspect requires the choice of a regularization parameter $\lambda > 0$ to be used in algorithm 1 (which is called in algorithms 2 and 4). In our experience, the theoretical choices $\lambda = 2\sigma\surd\log\{p\log(n)\}$ and $\lambda = 4\sigma\surd\log(np)$ used in theorems 1 and 2 produce consistent estimators as predicted by the theory but are slightly conservative, and in practice we recommend the choice $\lambda = \sigma\surd[2^{-1}\log\{p\log(n)\}]$ in both cases. Fig. 2 illustrates the dependence of the performance of our algorithm on the regularization parameter and reveals in this case (as in the other examples that we tried) that this choice of λ is sensible. In the implementation of our algorithm, we do not assume that the noise level σ is known, nor even that it is constant across different components. Instead, we estimate the error variance for each individual time series by using the median absolute deviation of first-order differences with scaling constant 1.05 for the normal distribution (Hampel, 1974). We then normalize each series by its estimated standard deviation and use the choices of λ given above with σ replaced by 1.

In step 2 of algorithm 2, we also have a choice between using $\mathcal{S} = \mathcal{S}_1$ and $\mathcal{S} = \mathcal{S}_2$. The following numerical experiment demonstrates the difference in performance of the algorithm for these two choices. We took $n = 500$, $p = 1000$, $k = 30$ and $\sigma^2 = 1$, with a single change point at $z = 200$. Table 5 shows the angles between the oracle projection direction and estimated projection directions by using both $\mathcal{S}_1$ and $\mathcal{S}_2$ as the signal level ϑ varies from 0.5 to 5.0. We have additionally reported the benchmark performance of the naive estimator by using the leading left singular vector of T, which illustrates that the convex optimization algorithms significantly improve the naive estimator by exploiting the sparsity structure. It can be seen that further relaxation from $\mathcal{S}_1$ to $\mathcal{S}_2$ incurs a relatively low cost in terms of the quality of estimation of the projection direction, but it offers great improvement in running time due to the closed form solution (see proposition 2 in the on-line supplement). Thus, even though the use of $\mathcal{S}_1$ remains a viable practical choice for offline data sets of moderate size, we use $\mathcal{S} = \mathcal{S}_2$ in the simulations that follow.

We compare the performance of the inspect algorithm with the following recently proposed methods for high dimensional change point estimation: the sparsified binary segmentation

Table 5. Angles between oracle projection direction v and estimated projection directions $\hat{v}_{\mathcal{S}_1}$ (using $\mathcal{S}_1$), $\hat{v}_{\mathcal{S}_2}$ (using $\mathcal{S}_2$) and $\hat{v}_{\max}$ (leading left singular vector of T), for various choices of ϑ†

ϑ	$\angle(\hat{v}_{\mathcal{S}_1}, v)$ (deg)	$\angle(\hat{v}_{\mathcal{S}_2}, v)$ (deg)	$\angle(\hat{v}_{\max}, v)$ (deg)
0.5	75.3	75.7	83.4
1.0	60.2	61.7	77.2
1.5	44.6	46.8	64.8
2.0	32.1	34.4	57.1
2.5	24.0	26.5	51.5
3.0	19.7	21.7	47.4
3.5	15.9	18.1	44.5
4.0	12.6	15.2	40.8
4.5	10.0	12.2	38.1
5.0	7.7	10.2	35.2

†Each reported value is averaged over 100 repetitions. Other simulation parameters: $n = 500$, $p = 1000$, $k = 30$, $z = 200$ and $\sigma^2 = 1$.

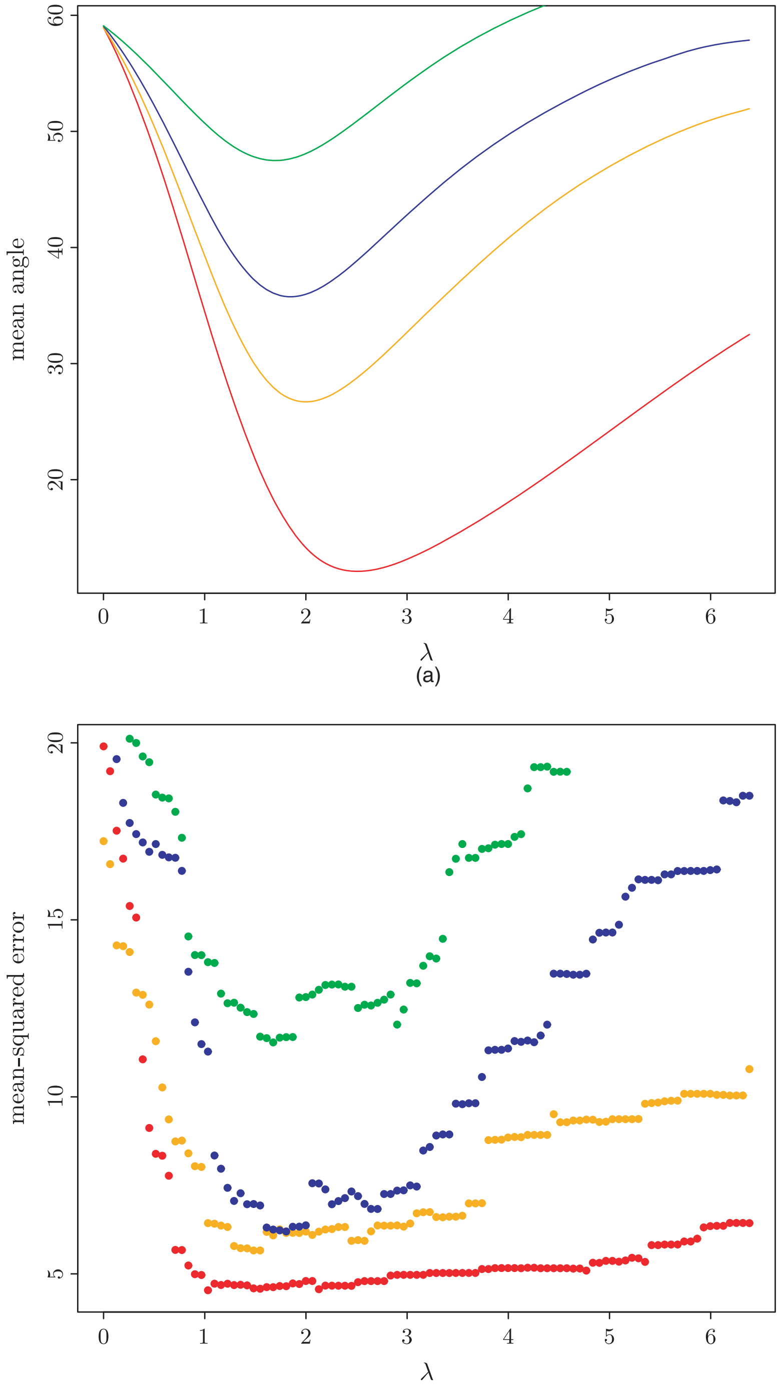


Fig. 2. Dependence of estimation performance on λ: (a) mean angle in degrees between the estimated projection direction and oracle projection direction over 100 experiments; (b) mean-squared error of the estimated change point location over 100 experiments ($n = 1000$, $p = 500$, $k = 3$ (red) or 10 (orange) or 22 (blue) or 100 (green), $z = 400$, $\vartheta = 1$ and $\sigma^2 = 1$; for these parameters, our choice of λ is $\sigma\sqrt{[2^{-1}\log\{p\log(n)\}]} \approx 2.02$)

algorithm sbs (Cho and Fryzlewicz, 2015), the double-CUSUM algorithm dc of Cho (2016), the scan-statistic-based algorithm scan derived from the work of Enikeeva and Harchaoui (2014), the l_∞ CUSUM aggregation algorithm agg_∞ of Jirak (2015) and the l_2 CUSUM aggregation algorithm agg_2 of Horváth and Hušková (2012). We remark that the latter three works primarily concern the test for the existence of a change point. The relevant test statistics can be naturally modified into a change point location estimator, though we note that optimal testing procedures may not retain their optimality for the estimation problem. Each of these methods can be extended to a multiple-change-point estimation algorithm via a wild binary segmentation scheme in a similar way to our algorithm, in which the termination criterion is chosen by fivefold cross-validation. Whenever tuning parameters are required in running these algorithms, we adopt the choices that were suggested by their authors in the relevant references.

5.1. Single-change-point estimation

All algorithms in our simulation study are top-down algorithms in the sense that their multiple-change-point procedure is built on a single-change-point estimation submodule, which is used to locate recursively all change points via a (wild) binary segmentation scheme. It is therefore instructive first to compare their performance in the single-change-point estimation task. Our simulations were run for $n, p \in \{500, 1000, 2000\}$, $k \in \{3, \lceil p^{1/2} \rceil, 0.1p, p\}$, $z = 0.4n$, $\sigma^2 = 1$ and $\vartheta = 0.8$, with $\theta \propto (1, 2^{-1/2}, \ldots, k^{-1/2}, 0, \ldots, 0)^{\mathrm{T}} \in \mathbb{R}^p$. For definiteness, we let the n columns of X be independent, with the leftmost z columns drawn from $N_p(0, \sigma^2 I_p)$ and the remaining columns drawn from $N_p(\theta, \sigma^2 I_p)$. To avoid the influence of different threshold levels on the performance of the algorithms and to focus solely on their precision of estimation, we assume that the existence of a single change point is known *a priori* and we make all algorithms output their estimate of its location; estimation of the number of change points in a multiple-change-point setting is studied in Section 5.3 below. Table 6 compares the performance of inspect and other competing algorithms under various parameter settings. All algorithms were run on the same data matrices and the root-mean-squared estimation error over 1000 repetitions is reported. Although, in the interests of brevity, we report the root-mean-squared estimation error only for $\vartheta = 0.8$, simulation results for other values of ϑ were qualitatively similar. We also remark that the four choices for the parameter k correspond to constant or logarithmic sparsity, polynomial sparsity and two levels of non-sparse settings. In addition to comparing the practical algorithms, we also computed the change point estimator based on the oracle projection direction (which of course is typically unknown); the performance of this oracle estimator depends only on n, z, ϑ and σ^2 (and not on k or p), and the corresponding root-mean-squared errors in Table 6 were 10.0, 8.1 and 7.8 when $(n, z, \vartheta, \sigma^2) = (500, 200, 0.8, 1), (1000, 400, 0.8, 1), (2000, 800, 0.8, 1)$ respectively. Thus the performance of our inspect algorithm is very close to that of the oracle estimator when k is small, as predicted by our theory.

As a graphical illustration of the performance of the various methods, Fig. 3 displays density estimates of their estimated change point locations in three settings. One difficulty in presenting such estimates with kernel density estimators is the fact that different algorithms would require different choices of bandwidth, and these would need to be locally adaptive, because of the relatively sharp peaks. To avoid the choice of bandwidth skewing the visual representation, we therefore use the log-concave maximum likelihood estimator for each method (e.g. Dümbgen and Rufibach (2009) and Cule *et al.* (2010)), which is both locally adaptive and tuning parameter free.

It can be seen from Table 6 and Fig. 3 that inspect has extremely competitive performance for the single-change-point estimation task. In particular, despite the fact that it is designed for

Table 6. Root-mean-squared error for inspect, dc, sbs, scan, agg_2 and agg_∞ in single-change-point estimation†

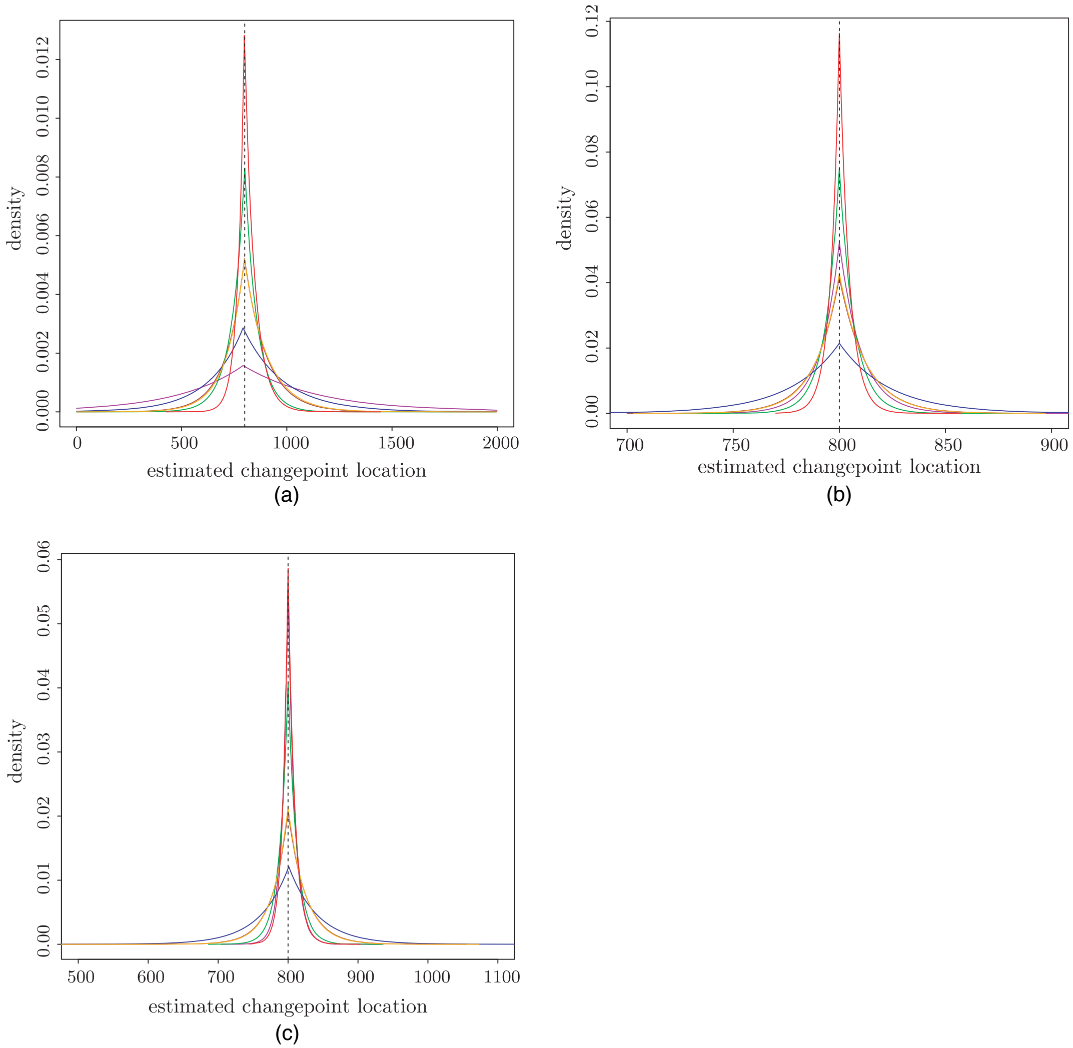
n	p	k	z	*Root-mean-squared errors for the following methods:*					
				inspect	*dc*	*sbs*	*scan*	*agg*$_2$	*agg*$_\infty$
500	500	3	200	*11.2*	22.2	72.7	11.6	115.9	22.4
500	500	22	200	*31.0*	80.8	87.1	65.7	113.2	83.1
500	500	50	200	*35.3*	105.9	102.9	86.8	112.7	107.9
500	500	500	200	*48.8*	147.7	129.6	120.0	114.6	150.8
500	1000	3	200	*13.0*	21.3	83.6	14.3	145.6	19.6
500	1000	32	200	*34.9*	104.6	114.9	95.0	144.9	107.5
500	1000	100	200	*45.0*	124.8	132.0	122.9	145.3	133.6
500	1000	1000	200	*55.0*	140.4	146.5	146.8	144.2	159.5
500	2000	3	200	*18.4*	56.0	99.4	26.4	163.0	26.6
500	2000	45	200	*43.5*	152.3	133.8	126.8	164.9	132.6
500	2000	200	200	*52.8*	159.1	151.6	150.6	163.2	158.4
500	2000	2000	200	*59.6*	162.1	162.4	166.1	163.0	176.0
1000	500	3	400	*8.4*	12.5	101.1	8.6	65.4	13.9
1000	500	22	400	*14.1*	44.2	60.6	18.7	66.7	44.4
1000	500	50	400	*19.7*	61.5	72.1	24.7	66.7	62.4
1000	500	500	400	*36.8*	137.8	114.8	77.4	72.8	142.6
1000	1000	3	400	9.5	14.6	117.2	*9.0*	154.9	15.0
1000	1000	32	400	*20.7*	61.1	83.6	26.4	150.1	57.2
1000	1000	100	400	*33.1*	101.0	122.0	59.2	158.3	106.4
1000	1000	1000	400	*57.7*	159.9	186.3	145.2	152.7	195.2
1000	2000	3	400	10.8	15.4	132.9	*10.3*	232.8	15.5
1000	2000	45	400	*29.6*	121.0	137.0	39.1	237.5	73.4
1000	2000	200	400	*47.4*	176.8	187.7	123.6	235.4	158.2
1000	2000	2000	400	*67.2*	219.6	240.0	210.3	233.4	245.8
2000	500	3	800	*8.6*	15.5	159.7	*8.6*	22.6	15.5
2000	500	22	800	*12.4*	31.2	48.7	17.0	25.9	32.1
2000	500	50	800	*14.6*	39.6	57.7	20.4	25.3	38.6
2000	500	500	800	*23.9*	72.7	86.1	35.6	25.1	71.8
2000	1000	3	800	*8.1*	14.2	178.3	8.3	42.6	14.4
2000	1000	32	800	*12.5*	36.1	58.7	16.9	40.6	38.2
2000	1000	100	800	*17.0*	46.7	75.8	24.6	40.0	47.3
2000	1000	1000	800	*31.0*	89.0	111.2	45.4	39.9	91.0
2000	2000	3	800	9.3	15.9	215.7	*9.0*	143.6	16.1
2000	2000	45	800	*16.7*	35.8	100.7	21.3	152.5	39.2
2000	2000	200	800	*25.6*	56.7	126.5	32.0	151.8	59.1
2000	2000	2000	800	*48.4*	107.9	208.0	66.1	150.6	153.5

†The smallest root-mean-squared error is given in italics. Other parameters: $\vartheta = 0.8$ and $\sigma^2 = 1$.

estimation of sparse change points, inspect performs relatively well even when $k = p$ (i.e. when the signal is highly non-sparse).

5.2. *Other data-generating mechanisms*

We now extend the ideas of Section 5.1 by investigating empirical performance under several other data-generating mechanisms. Recall that the noise matrix is $W = (W_{j,t}) := X - \boldsymbol{\mu}$ and we define $W_1, \ldots, W_n$ to be the column vectors of W. In models M_{unif} and M_{exp}, we replace Gaussian noise by $W_{j,t} \sim^{\text{IID}} \text{Unif}[-\surd 3\sigma, \surd 3\sigma]$ and $W_{j,t} \sim^{\text{IID}} \text{Exp}(\sigma) - \sigma$ respectively. We note that the correct Hampel scaling constants are approximately 0.99 and 1.44 in these two cases, though we continue to use the constant 1.05 for normally distributed data. In model



(a)

(b)

(c)

Fig. 3. Estimated densities of location of change point estimates by inspect (——), dc (——), sbs (——), scan (——), agg$_2$ (——) and agg$_\infty$ (——): (a) $(n,p,k,z,\vartheta,\sigma^2)=(2000,1000,32,800,0.5,1)$; (b) $(n,p,k,z,\vartheta,\sigma^2)=(2000,1000,32,800,1,1)$; (c) $(n,p,k,z,\vartheta,\sigma^2)=(2000,1000,1000,800,1,1)$

$M_{\mathrm{cs,loc}}(\rho)$, we allow the noise to have a short-range cross-sectional dependence by sampling $W_1,\ldots,W_n \sim^{\mathrm{IID}} N_p(0,\Sigma)$ for $\Sigma := (\rho^{|j-j'|})_{j,j'}$. In model $M_{\mathrm{cs}}(\rho)$, we extend this to global cross-sectional dependence by sampling $W_1,\ldots,W_n \sim^{\mathrm{IID}} N_p(0,\Sigma)$ for $\Sigma := (1-\rho)I_p + (\rho/p)\mathbf{1}_p\mathbf{1}_p^{\mathrm{T}}$, where $\mathbf{1}_p \in \mathbb{R}^p$ is an all-1 vector. In model $M_{\mathrm{temp}}(\rho)$, we consider an auto-regressive AR(1) temporal dependence in the noise by first sampling $W'_{j,t} \sim^{\mathrm{IID}} N(0,\sigma^2)$ and then setting $W_{j,1} := W'_{j,1}$ and $W_{j,t} := \rho^{1/2}W_{j,t-1} + (1-\rho)^{1/2}W'_{j,t}$ for $2 \leqslant t \leqslant n$. In $M_{\mathrm{async}}(L)$, we model asynchronous change point location in the signal co-ordinates by drawing change point locations for individual co-ordinates independently from a uniform distribution on $\{z-L,\ldots,z+L\}$. We report the performance of the various algorithms in the parameter setting $n=2000$, $p=1000$, $k=32$, $z=800$, $\vartheta=0.25$ and $\sigma^2=1$ in Table 7. It can be seen that inspect is robust to spatial dependence structures, noise misspecification and moderate temporal dependence, though its performance

Table 7. Root-mean-squared error for inspect, dc, sbs, scan, agg_2 and agg_∞ in single-change-point estimation, under different data-generating mechanisms

Model	*n*	*p*	*k*	*z*	ϑ	*Root-mean-squared errors for the following methods:*					
						inspect	*dc*	*sbs*	*scan*	agg_2	agg_∞
M_{unif}	2000	1000	32	800	1.5	*2.7*	9.6	17.1	4.9	4.3	10.2
M_{exp}	2000	1000	32	800	1.5	*2.6*	9.6	42.6	5.0	4.7	9.6
$M_{\mathrm{cs\_loc}}(0.2)$	2000	1000	32	800	1.5	*3.5*	9.7	19.2	7.0	5.4	9.8
$M_{\mathrm{cs\_loc}}(0.5)$	2000	1000	32	800	1.5	*5.8*	9.7	24.6	8.7	9.3	9.6
$M_{\mathrm{cs}}(0.5)$	2000	1000	32	800	1.5	*1.5*	7.7	14.9	3.0	3.6	6.7
$M_{\mathrm{cs}}(0.9)$	2000	1000	32	800	1.5	*2.7*	9.9	18.6	4.7	4.7	9.6
$M_{\mathrm{temp}}(0.1)$	2000	1000	32	800	1.5	*6.1*	20.3	102.8	9.4	10.9	20.2
$M_{\mathrm{temp}}(0.3)$	2000	1000	32	800	1.5	*30.1*	32.4	276.4	38.8	38.2	34.8
$M_{\mathrm{temp}}(0.5)$	2000	1000	32	800	1.5	85.1	*57.0*	379.6	61.8	83.4	76.6
$M_{\mathrm{temp}}(0.7)$	2000	1000	32	800	1.5	243.6	*177.3*	456.7	189.0	239.5	190.5
$M_{\mathrm{async}}(10)$	2000	1000	32	800	1.5	*5.8*	11.5	18.5	7.8	7.0	11.3

deteriorates slightly relatively to other methods in the presence of strong temporal correlation, apparently due to slight under-regularization in these latter settings.

5.3. Multiple-change-point estimation

The use of the ‘burn-off’ parameter β in algorithm 4 was mainly to facilitate our theoretical analysis. In our simulations, we found that taking $\beta=0$ rarely resulted in the change point being estimated more than once, and we therefore recommend setting $\beta=0$ in practice, unless prior knowledge of the distribution of the change points suggests otherwise. To choose ξ in the multiple-change-point estimation simulation studies, for each (n, p), we first applied inspect to 1000 data sets drawn from the null model with no change point and took ξ to be the largest value of $\bar{T}_{\max}$ from algorithm 2. We also set $Q=1000$.

We consider the simulation setting where $n=2000$, $p=200$, $k=40$, $\sigma^2=1$ and $z=(500, 1000, 1500)$. Define $\vartheta^{(i)} := \|\theta^{(i)}\|_2$ to be the signal strength at the ith change point. We set $(\vartheta^{(1)}, \vartheta^{(2)}, \vartheta^{(3)}) = (\vartheta, 2\vartheta, 3\vartheta)$ and take $\vartheta \in \{0.4, 0.6\}$ to see the performance of the algorithms at various signal strengths. We also considered different levels of overlap between the co-ordinates in which the three changes in mean structure occur: in the *complete-overlap* case, changes occur in the same k co-ordinates at each change point; in the *half-overlap* case, the changes occur in co-ordinates

$$\frac{i-1}{2}k+1, \ldots, \frac{i+1}{2}k$$

for $i=1,2,3$; in the *no-overlap* case, the changes occur in disjoint sets of co-ordinates. Table 8 summarizes the results. We report both the frequency counts of the number of change points detected over 100 runs (all algorithms were compared over the same set of randomly generated data matrices) and two quality measures of the location of change points. In particular, since change point estimation can be viewed as a special case of classification, the quality of the estimated change points can be measured by the adjusted Rand index ARI of the estimated segmentation against the truth (Rand, 1971; Hubert and Arabie, 1985). We report both the average ARI over all runs and the percentage of runs for which a particular method attains the largest ARI among the six. Fig. 4 gives a pictorial representation of the results for one

Table 8. Multiple-change-point simulation results†

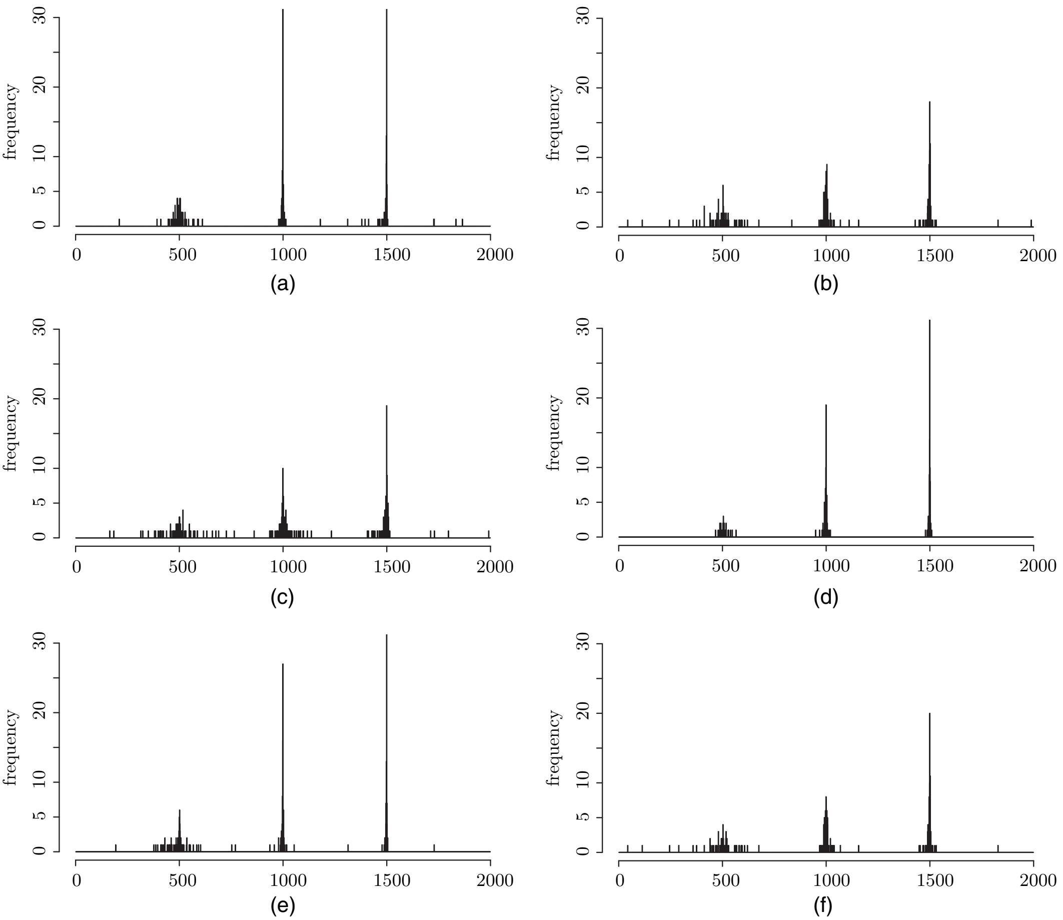
$(\vartheta^{(1)},\vartheta^{(2)},\vartheta^{(3)})$	*Method*	*Results for the following values of $\hat{\nu}$:*						*ARI*	*% best*
		0	*1*	*2*	*3*	*4*	*5*		
(0.6, 1.2, 1.8)	inspect	0	0	20	72	8	0	0.90	55
	dc	0	0	21	54	23	2	0.85	22
	sbs	0	0	12	64	22	2	0.86	15
	scan	0	0	72	27	1	0	0.77	8
	agg_2	0	0	18	73	8	1	0.87	1
	agg_∞	0	0	29	57	13	1	0.83	17
(0.4, 0.8, 1.2)	inspect	0	0	62	34	4	0	0.74	50
	dc	0	0	62	32	5	1	0.69	19
	sbs	0	0	54	44	1	1	0.70	21
	scan	0	2	95	3	0	0	0.68	19
	agg_2	0	0	81	17	2	0	0.71	2
	agg_∞	0	0	68	29	3	0	0.68	8
(0.6, 1.2, 1.8)	inspect	0	0	20	70	10	0	0.90	51
	dc	0	0	24	58	17	1	0.87	27
	sbs	0	0	17	61	17	5	0.85	11
	scan	0	0	74	26	0	0	0.78	15
	agg_2	0	0	30	67	2	1	0.86	3
	agg_∞	0	0	32	58	9	1	0.85	15
(0.4, 0.8, 1.2)	inspect	0	0	65	31	4	0	0.73	44
	dc	0	0	73	25	2	0	0.70	18
	sbs	0	0	65	29	6	0	0.68	16
	scan	0	2	96	2	0	0	0.70	29
	agg_2	0	0	83	14	3	0	0.71	5
	agg_∞	0	0	82	17	1	0	0.69	12
(0.6, 1.2, 1.8)	inspect	0	0	19	71	9	1	0.90	55
	dc	0	0	28	53	17	2	0.85	22
	sbs	0	0	18	67	14	1	0.85	14
	scan	0	0	74	26	0	0	0.78	14
	agg_2	0	0	23	66	10	1	0.87	0
	agg_∞	0	0	32	58	9	1	0.85	10
(0.4, 0.8, 1.2)	inspect	0	0	66	30	4	0	0.74	50
	dc	0	0	75	23	2	0	0.70	18
	sbs	0	0	62	30	7	1	0.69	11
	scan	0	1	98	1	0	0	0.70	29
	agg_2	0	0	86	12	2	0	0.72	5
	agg_∞	0	0	82	15	3	0	0.70	7

†The top, middle and bottom blocks refer to the complete-, half- and no-overlap settings respectively. Other simulation parameters: $n=2000$, $p=200$, $k=40$, $z=(500, 1000, 1500)$ and $\sigma^2=1$.

particular collection of parameter settings. Again, we find that the performance of inspect is very encouraging on all performance measures, though we remark that agg_2 is also competitive, and scan tends to output the fewest false positive results.

5.4. Real data application

We study the comparative genomic hybridization microarray data set from Bleakley and Vert (2011), which is available in the `ecp` R package (James and Matteson, 2015). Comparative genomic hybridization is a technique that allows detection of chromosomal copy number abnormality by comparing the fluorescence intensity levels of DNA fragments from a test sample



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 4. Histograms of estimated change point locations by (a) inspect, (b) dc, (c) sbs, (d) scan, (e) agg$_2$ and (f) agg$_\infty$ in the half-overlap case (parameter settings: $n=2000$, $p=200$, $k=40$, $z=(500,1000,1500)$, $(\vartheta^{(1)},\vartheta^{(2)},\vartheta^{(3)})=(0.6,1.2,1.8)$, $\sigma^2=1$)

and a reference sample. This data set contains (test-to-reference) log-intensity-ratio measurements of 43 individuals with bladder tumours at 2215 different loci on their genome. The log-intensity-ratios for the first 10 individuals are plotted in Fig. 5. Whereas some of the copy number variations are specific to one individual, some copy number abnormality regions (e.g. between loci 2044 and 2143) are shared across several individuals and are more likely to be disease related. The inspect algorithm aggregates the changes in different individuals and estimates the start and end points of copy number changes. Because of the large number of individual-specific copy number changes and the presence of measurement outliers, direct application of inspect with the default threshold level identifies 254 change points. However, practitioners can use the associated $\bar{T}^{[q_0]}_{\max}$-score to identify the most significant changes. The 30 most significant identified change points are plotted as red broken lines in Fig. 5.

6. Extensions: temporal or spatial dependence

In this section, we explore how our method and its analysis can be extended to handle more

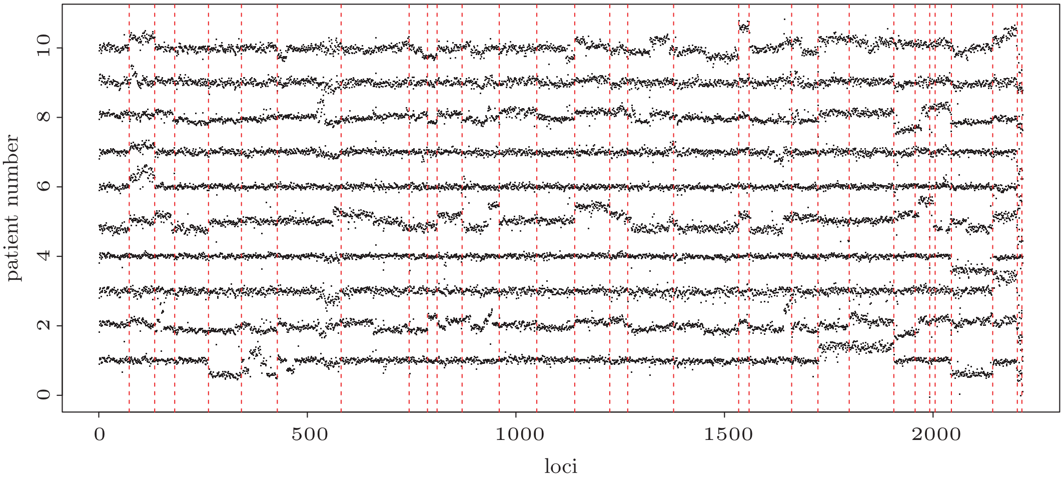


Fig. 5. Log-intensity-ratio measurements of microarray data (only the first 10 patients are shown): ¦, change points estimated by using all patients in the data set

realistic streaming data settings where our data exhibit temporal or spatial dependence. For simplicity, we focus on the single-change-point case and assume the same mean structure for $\boldsymbol{\mu} = \mathbb{E}(X)$ as described in Section 2, in particular expressions (2), (3), (4), (5) and (6).

6.1. Temporal dependence

A natural way of relaxing the assumption of independence of the columns of our data matrix is to assume that the noise vectors $W_1, \ldots, W_n$ are stationary. Writing $K(u) := \mathrm{cov}(W_t, W_{t+u})$, we assume here that $W = (W_1, \ldots, W_n)$ forms a centred, stationary Gaussian process with covariance function K. As we are mainly interested in the temporal dependence in this subsection, we assume that each component time series evolves independently, so that $K(u)$ is a diagonal matrix for every u. Further, writing $\sigma^2 := \|K(0)\|_{\mathrm{op}}$, we shall assume that the dependence is short ranged, in the sense that

$$\left\| \sum_{u=0}^{n-1} K(u) \right\|_{\mathrm{op}} \leqslant B\sigma^2 \tag{18}$$

for some universal constant $B > 0$. In this case, the oracle projection direction is still $v := \theta/\|\theta\|_2$ and our inspect algorithm does not require any modification. In terms of its performance in this context, we have the following result.

Theorem 3. Suppose that $\sigma, B > 0$ are known. Let $\hat{z}$ be the output of algorithm 3 with input X and $\lambda := \sigma\surd\{8B\log(np)\}$. There are universal constants $C, C' > 0$ such that, if $n \geqslant 12$ is even, z is even and

$$\frac{C\sigma}{\vartheta\tau}\sqrt{\left\{\frac{kB\log(np)}{n}\right\}} \leqslant 1, \tag{19}$$

then

$$\mathbb{P}\left\{\frac{1}{n}|\hat{z} - z| \leqslant \frac{C'\sigma^2 B\log(n)}{n\vartheta^2}\right\} \geqslant 1 - \frac{12}{n}.$$

6.2. Spatial dependence

Now consider the case where we have spatial dependence between the different co-ordinates of the data stream. More specifically, suppose that the noise vectors satisfy $W_1, \ldots, W_n \sim^{\mathrm{IID}} N_p(0, \Sigma)$, for some positive definite matrix $\Sigma \in \mathbb{R}^{p \times p}$. This turns out to be a more complicated setting, where our initial algorithm requires modification. To see this, observe now that, for $a \in \mathbb{S}^{p-1}$,

$$a^{\mathrm{T}} X_t \sim N(a^{\mathrm{T}} \mu_t, a^{\mathrm{T}} \Sigma a).$$

It follows that the oracle projection direction in this case is

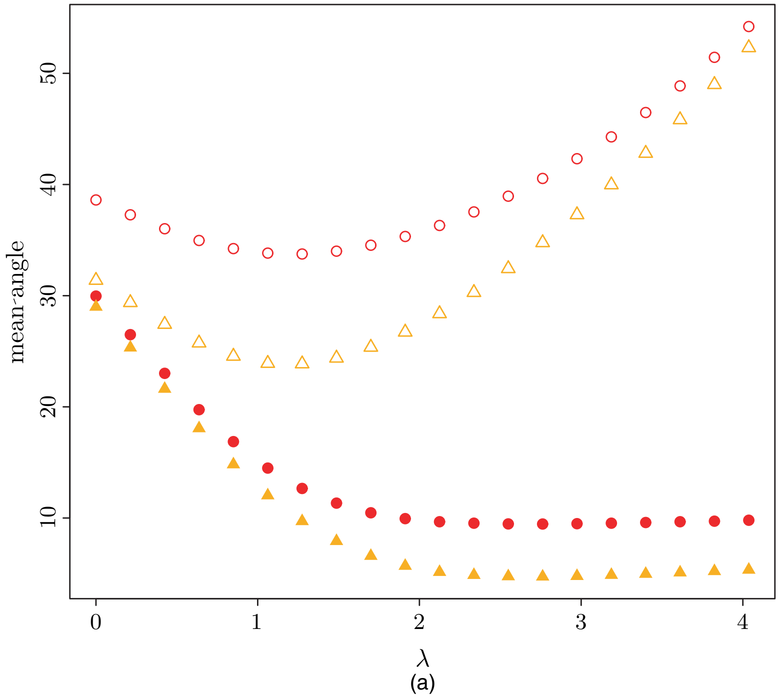
$$v_{\mathrm{proj}} := \underset{a \in \mathbb{S}^{p-1}}{\arg\max} \frac{|a^{\mathrm{T}} \theta|}{\sqrt{(a^{\mathrm{T}} \Sigma a)}} = \Sigma^{-1/2} \underset{b \in \mathbb{S}^{p-1}}{\arg\max} |b^{\mathrm{T}} \Sigma^{-1/2} \theta| = \frac{\Sigma^{-1} \theta}{\|\Sigma^{-1} \theta\|_2}.$$

If $\hat{\Theta}$ is an estimator of the precision matrix $\Theta := \Sigma^{-1}$, and $\hat{v}$ is a leading left singular vector of $\hat{M}$ as computed in step 3 of algorithm 2, then we can estimate the oracle projection direction by $\hat{v}_{\mathrm{proj}} := \hat{\Theta} \hat{v} / \|\hat{\Theta} \hat{v}\|_2$. The sample splitting version of this algorithm is therefore given in algorithm 5 in Table 9. Lemma 16 in the on-line supplement allows us to control $\sin\{\angle(\hat{v}_{\mathrm{proj}}, v_{\mathrm{proj}})\}$ in terms of $\sin\{\angle(\hat{v}, v)\}$ and $\|\hat{\Theta} - \Theta\|_{\mathrm{op}}$, as well as the extreme eigenvalues of Θ. Since proposition 1 does not rely on the independence of the different co-ordinates, it can still be used to control $\sin\{\angle(\hat{v}, v)\}$. In general, controlling $\|\hat{\Theta} - \Theta\|_{\mathrm{op}}$ in high dimensional cases requires assumptions of additional structure on Θ (or, equivalently, on Σ). For convenience of our theoretical analysis, we assume that we have access to observations $W'_1, \ldots, W'_m \sim^{\mathrm{IID}} N_p(0, \Sigma)$, independent of $X^{(2)}$, with which we can estimate Θ. In practice, if a lower bound on τ were known, we could take $W'_1, \ldots, W'_m$ to be scaled, disjoint first-order differences of the observations in $X^{(1)}$ that are within $n_1 \tau$ of the end points of the data stream; more precisely, we can let $W'_t := 2^{1/2}(X^{(1)}_{2t} - X^{(1)}_{2t-1})$ for $t = 1, \ldots, \lfloor n_1 \tau / 2 \rfloor$ and $W'_{\lfloor n_1 \tau / 2 \rfloor + t} := 2^{1/2}(X^{(1)}_{n_1 - 2t} - X^{(1)}_{n_1 - 2t + 1})$, so that $m = 2 \lfloor n_1 \tau / 2 \rfloor$. In fact, lemmas 17 and 18 in the on-line supplement indicate that, at least for certain dependence structures, the operator norm error in estimation of Θ is often negligible by comparison with $\sin\{\angle(\hat{v}, v)\}$, so a fairly crude lower bound on τ would often suffice.

Theoretical guarantees on the performance of the spatially dependent version of the inspect algorithm in illustrative examples of both local and global dependence structures are provided in theorem 4 in the on-line supplement. The main message of these results is that, provided that the dependence is not too strong, and we have a reasonable estimate of Θ, we attain the same rate of convergence as when there is no spatial dependence. However, theorem 4 also quantifies the

Table 9. Algorithm 5: pseudocode for a sample splitting variant of algorithm 2 for spatially dependent data

```
Input: $X \in \mathbb{R}^{p \times n}$, $\lambda > 0$
Step 1: perform the CUSUM transformation $T^{(1)} \leftarrow \mathcal{T}(X^{(1)})$ and $T^{(2)} \leftarrow \mathcal{T}(X^{(2)})$
Step 2: use algorithm 1 or equation (14) (with inputs $T^{(1)}$ and $\lambda$ in either case) to solve for $\hat{M}^{(1)} \in \arg\max_{M \in \mathcal{S}} \{\langle T^{(1)}, M \rangle - \lambda \|M\|_1\}$ with $\mathcal{S} = \{M \in \mathbb{R}^{p \times (n_1 - 1)} : \|M\|_* \leqslant 1\}$ or $\{M \in \mathbb{R}^{p \times (n_1 - 1)} : \|M\|_2 \leqslant 1\}$
Step 3: find $\hat{v}^{(1)} \in \arg\max_{\tilde{v} \in \mathbb{S}^{p-1}} \|(\hat{M}^{(1)})^{\mathrm{T}} \tilde{v}\|_2$
Step 4: let $\hat{\Theta}^{(1)} = \hat{\Theta}^{(1)}(X^{(1)})$ be an estimator of $\Theta$: let $\hat{v}^{(1)}_{\mathrm{proj}} \leftarrow \hat{\Theta}^{(1)} \hat{v}^{(1)}$
Step 5: let $\hat{z} \in 2 \arg\max_{1 \leqslant t \leqslant n_1 - 1} |(\hat{v}^{(1)}_{\mathrm{proj}})^{\mathrm{T}} T^{(2)}_t|$, and set $\bar{T}_{\max} \leftarrow |(\hat{v}^{(1)}_{\mathrm{proj}})^{\mathrm{T}} T^{(2)}_{\hat{z}/2}|$
Output: $\hat{z}, \bar{T}_{\max}$
```



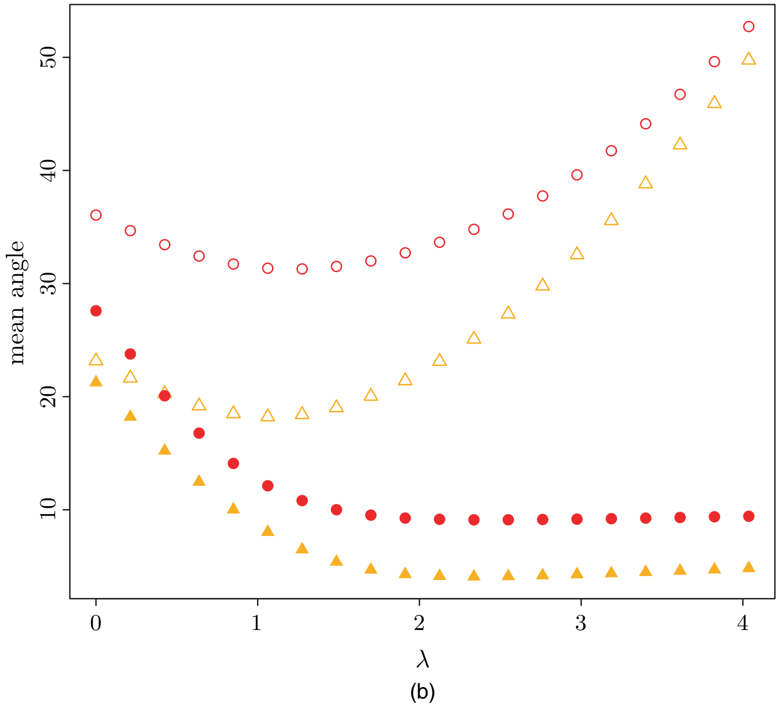


Fig. 6. Mean angle between the estimated projection direction and the optimal projection direction v_{proj} over 100 experiments ($n=1000$, $p=500$, $k=10$ (●, ▲) or $k=100$ (○, △), $z=400$, $\vartheta=3$, (a) $\Sigma=(\Sigma_{i,j})=2^{-|i-j|}$ or (b) $\Sigma=I_p+\mathbf{1}_p\mathbf{1}_p^{\mathrm{T}}/2$): ●, ○, vanilla inspect algorithm; ▲, △, algorithm 5

way in which this rate of convergence deteriorates as the dependence approaches the boundary of its range.

In Fig. 6, we compare the performances of the ‘vanilla’ inspect algorithm (algorithm 3) and algorithm 5 on simulated data sets with local and spatial dependence structures. We observe that algorithm 5 offers improved performance across all values of λ considered by accounting for the spatial dependence, as suggested by our theoretical arguments.

Acknowledgements

The research of both authors is supported by the second author’s Engineering and Physical Sciences Research Council Fellowship and a grant from the Leverhulme Trust. The second author is also supported by an Engineering and Physical Sciences Research Council Programme grant and the Alan Turing Institute.

Appendix A: Proofs of main results

A.1. Proof (of proposition 1)

We note that the matrix A as defined in Section 3 has rank 1, and its only non-zero singular value is $\|\theta\|_2\|\gamma\|_2$. By proposition 7 in the on-line supplement, on the event $\Omega_* := \{\|E\|_\infty \leqslant \lambda\}$, we have

$$\sin\{\angle(\hat{v}, v)\} \leqslant \frac{8\lambda\surd(kn)}{\|\theta\|_2\|\gamma\|_2}.$$

By definition, $\|\theta\|_2 \geqslant \vartheta$, and, by lemma 8 in the on-line supplement, $\|\gamma\|_2 \geqslant \frac{1}{4}n\tau$. Thus, $\sin\{\angle(\hat{v}, v)\} \leqslant 32\lambda\surd k/(\vartheta\tau\surd n)$ on Ω_*. It remains to verify that $\mathbb{P}(\Omega_*^{\mathrm{c}}) \leqslant 4\{p\log(n)\}^{-1/2}$ for $n \geqslant 6$. By lemma 9 in the on-line supplement,

$$\begin{aligned}\mathbb{P}(\|E\|_\infty \geqslant 2\sigma\surd[\log\{p\log(n)\}]) &\leqslant 2\sqrt{\left(\frac{2}{\pi}\right)}\,p\lceil\log(n)\rceil\surd\log\{p\log(n)\}\left[1+\frac{1}{\log\{p\log(n)\}}\right]\{p\log(n)\}^{-2}\\ &\leqslant 6\{p\log(n)\}^{-1}\surd\log\{p\log(n)\} \leqslant 4\{p\log(n)\}^{-1/2},\end{aligned} \tag{20}$$

as desired.

A.2. Proof (of theorem 1)

Recall the definition of $X^{(2)}$ in expression (16) and the definition $T^{(2)} := \mathcal{T}(X^{(2)})$. Define similarly $\boldsymbol{\mu}^{(2)} = (\mu_1^{(2)}, \ldots, \mu_{n_1}^{(2)}) \in \mathbb{R}^{p\times n_1}$ and a random $W^{(2)} = (W_1^{(2)}, \ldots, W_{n_1}^{(2)})$ taking values in $\mathbb{R}^{p\times n_1}$ by $\mu_t^{(2)} := \mu_{2t}$ and $W_t^{(2)} = W_{2t}$; now let $A^{(2)} := \mathcal{T}(\boldsymbol{\mu}^{(2)})$ and $E^{(2)} := \mathcal{T}(W^{(2)})$. Furthermore, we write $\bar{X} := (\hat{v}^{(1)})^{\mathrm{T}}X^{(2)}$, $\bar{\mu} := (\hat{v}^{(1)})^{\mathrm{T}}\boldsymbol{\mu}^{(2)}$, $\bar{W} := (\hat{v}^{(1)})^{\mathrm{T}}W^{(2)}$, $\bar{T} := (\hat{v}^{(1)})^{\mathrm{T}}T^{(2)}$, $\bar{A} := (\hat{v}^{(1)})^{\mathrm{T}}A^{(2)}$ and $\bar{E} := (\hat{v}^{(1)})^{\mathrm{T}}E^{(2)}$ for the one-dimensional projected images (as row vectors) of the corresponding p-dimensional quantities. We note that $\bar{T} = \mathcal{T}(\bar{X})$, $\bar{A} = \mathcal{T}(\bar{\mu})$ and $\bar{E} = \mathcal{T}(\bar{W})$.

Now, conditionally on $\hat{v}^{(1)}$, the random variables $\bar{X}_1, \ldots, \bar{X}_{n_1}$ are independent, with

$$\bar{X}_t|\hat{v}^{(1)} \sim N(\bar{\mu}_t, \sigma^2),$$

and the row vector $\bar{\mu}$ undergoes a single change at $z^{(2)} := z/2$ with magnitude of change

$$\bar{\theta} := \bar{\mu}_{z^{(2)}+1} - \bar{\mu}_{z^{(2)}} = (\hat{v}^{(1)})^{\mathrm{T}}\theta.$$

Finally, let $\hat{z}^{(2)} \in \arg\max_{1\leqslant t\leqslant n_1-1}|\bar{T}_t|$, so the first component of the output of the algorithm is $\hat{z} = 2\hat{z}^{(2)}$. Consider the set

$$\Upsilon := \{\tilde{v} \in \mathbb{S}^{p-1} : \angle(\tilde{v}, v) \leqslant \pi/6\}.$$

By condition (17) in the statement of theorem 1 and proposition 1,

$$\mathbb{P}(\hat{v}^{(1)} \in \Upsilon) \geqslant 1 - 4\{p\log(n_1)\}^{-1/2}. \tag{21}$$

Moreover, for $\hat{v}^{(1)} \in \Upsilon$, we have $\bar{\theta} \geqslant \sqrt{3}\vartheta/2$. Note also that $\hat{v}^{(1)}$ and $W^{(2)}$ are independent, so $\bar{W}$ has independent $N(0,\sigma^2)$ entries. Define $\lambda_1 := 3\sigma\sqrt{[\log\{\log(n_1)\}]}$. By lemma 9 in the on-line supplement, and the fact that $n \geqslant 12$, we have

$$\mathbb{P}(\|\bar{E}\|_\infty \geqslant \lambda_1) \leqslant \sqrt{(2/\pi)}\lceil\log(n_1)\rceil\left[3\sqrt{\log\{\log(n_1)\}} + \frac{2}{3\sqrt{\log\{\log(n_1)\}}}\right]\log(n_1)^{-9/2} \leqslant \log(n_1)^{-1}. \qquad (22)$$

Since $\bar{T} = \bar{A} + \bar{E}$, and since $(\bar{A}_t)_t$ and $(\bar{T}_t)_t$ are respectively maximized at $t = z^{(2)}$ and $t = \hat{z}^{(2)}$, we have on the event $\Omega_0 := \{\hat{v}^{(1)} \in \Upsilon, \|\bar{E}\|_\infty \leqslant \lambda_1\}$ that

$$\begin{aligned}\bar{A}_{z^{(2)}} - \bar{A}_{\hat{z}^{(2)}} &= (\bar{A}_{z^{(2)}} - \bar{T}_{z^{(2)}}) + (\bar{T}_{z^{(2)}} - \bar{T}_{\hat{z}^{(2)}}) + (\bar{T}_{\hat{z}^{(2)}} - \bar{A}_{\hat{z}^{(2)}})\\ &\leqslant |\bar{A}_{z^{(2)}} - \bar{T}_{z^{(2)}}| + |\bar{T}_{\hat{z}^{(2)}} - \bar{A}_{\hat{z}^{(2)}}| \leqslant 2\lambda_1.\end{aligned}$$

The row vector $\bar{A}$ has the explicit form

$$\bar{A}_t = \begin{cases} \sqrt{\left\{\dfrac{t}{n_1(n_1-t)}\right\}}(n_1 - z^{(2)})\bar{\theta}, & \text{if } t \leqslant z^{(2)},\\ \sqrt{\left(\dfrac{n_1-t}{n_1 t}\right)}z^{(2)}\bar{\theta}, & \text{if } t > z^{(2)}.\end{cases}$$

Hence, by lemma 12 in the on-line supplement, on the event Ω_0 we have that

$$\frac{|\hat{z}^{(2)} - z^{(2)}|}{n_1\tau} \leqslant \frac{3\sqrt{6}\lambda_1}{\bar{\theta}(n_1\tau)^{1/2}} = \frac{9\sqrt{6}\sigma}{\bar{\theta}}\sqrt{\left[\frac{\log\{\log(n_1)\}}{n_1\tau}\right]} \leqslant \frac{36\sigma}{\vartheta}\sqrt{\left[\frac{\log\{\log(n)\}}{n\tau}\right]}. \qquad (23)$$

Now define the event

$$\Omega_1 := \left\{\left|\sum_{r=1}^{s}\bar{W}_r - \sum_{r=1}^{t}\bar{W}_r\right| \leqslant \lambda_1\sqrt{|s-t|}, \quad \forall\, 0 \leqslant t \leqslant n_1, s \in \{0, z^{(2)}, n_1\}\right\}. \qquad (24)$$

From expression (23) and condition (17), provided that $C \geqslant 72$, we have $|\hat{z}^{(2)} - z^{(2)}| \leqslant n_1\tau/2$. We can therefore apply lemmas 11 and 12 in the on-line supplement and conclude that, on $\Omega_0 \cap \Omega_1$, we have

$$\begin{aligned}|\bar{E}_{z^{(2)}} - \bar{E}_{\hat{z}^{(2)}}| &\leqslant 2\sqrt{2}\lambda_1\sqrt{\left(\frac{|z^{(2)} - \hat{z}^{(2)}|}{n_1\tau}\right)} + 8\lambda_1\frac{|z^{(2)} - \hat{z}^{(2)}|}{n_1\tau},\\ \bar{A}_{z^{(2)}} - \bar{A}_{\hat{z}^{(2)}} &\geqslant \frac{2\bar{\theta}}{3\sqrt{6}}|z^{(2)} - \hat{z}^{(2)}|(n_1\tau)^{-1/2}.\end{aligned}$$

Since $\bar{T}_{z^{(2)}} \leqslant \bar{T}_{\hat{z}^{(2)}}$, we have that, on $\Omega_0 \cap \Omega_1$,

$$\begin{aligned}1 \leqslant \frac{|\bar{E}_{z^{(2)}} - \bar{E}_{\hat{z}^{(2)}}|}{\bar{A}_{z^{(2)}} - \bar{A}_{\hat{z}^{(2)}}} &\leqslant \frac{6\sqrt{3}\lambda_1}{\bar{\theta}|z^{(2)} - \hat{z}^{(2)}|^{1/2}} + \frac{12\sqrt{6}\lambda_1}{\bar{\theta}(n_1\tau)^{1/2}}\\ &\leqslant \frac{36\sqrt{2}\sigma}{\vartheta}\sqrt{\left[\frac{\log\{\log(n)\}}{|z - \hat{z}|}\right]} + \frac{144\sigma}{\vartheta}\sqrt{\left[\frac{\log\{\log(n)\}}{n\tau}\right]}.\end{aligned}$$

We conclude from condition (17) again, that on $\Omega_0 \cap \Omega_1$, for $C \geqslant 288$, we have

$$|\hat{z} - z| \leqslant C'\sigma^2\vartheta^{-2}\log\{\log(n)\}$$

for some universal constant $C' > 0$.

It remains to show that $\Omega_0 \cap \Omega_1$ has the desired probability. From expressions (21) and (22), as well as lemma 10 in the on-line supplement,

$$\mathbb{P}(\Omega_0^{\mathrm{c}} \cup \Omega_1^{\mathrm{c}}) \leqslant 4\{p\log(n_1)\}^{-1/2} + \log(n_1)^{-1} + 16\log(n_1)^{-5/4} \leqslant 4\{p\log(n_1)\}^{-1/2} + 17\{\log(n_1)\}^{-1}$$

as desired.

A.3. Proof (of theorem 3)

Writing $E^{(1)} := \mathcal{T}(W^{(1)})$ and $n_1 := n/2$, by lemma 15 in the on-line supplement and a union bound, we have that the event $\Omega_* := \{\|E^{(1)}\|_\infty \leqslant \lambda\}$ satisfies

$$\mathbb{P}(\Omega_*^c) = \mathbb{P}[\|E^{(1)}\|_\infty \geqslant \sigma\sqrt{\{8B\log(n_1 p)\}}] \leqslant (n_1 - 1)p \exp\{-2\log(n_1 p)\} \leqslant \frac{1}{n_1 p}.$$

Moreover, following the proof of proposition 1, on Ω_*,

$$\sin\{\angle(\hat{v}^{(1)}, v)\} \leqslant \frac{64\sqrt{2}\sigma\sqrt{\{kB\log(n_1 p)\}}}{\tau\vartheta\sqrt{n_1}} \leqslant \frac{1}{2},$$

provided that, in condition (19), we take the universal constant $C > 0$ sufficiently large. Now following the notation and proof of theorem 1, but using lemma 15 instead of lemma 9 in the on-line supplement, and writing $\lambda_1 := \sigma\sqrt{\{8B\log(n_1)\}}$, we have

$$\mathbb{P}(\|\bar{E}\|_\infty \geqslant \lambda_1) \leqslant (n_1 - 1)\exp\{-2\log(n_1)\} \leqslant \frac{1}{n_1}.$$

Similarly, using lemma 15 in the on-line supplement again instead of lemma 10, the event Ω_1 defined in expression (24) satisfies

$$\mathbb{P}(\Omega_1^c) \leqslant 4n_1 \exp\left(-\frac{\lambda_1^2}{4B\sigma^2}\right) \leqslant \frac{4}{n_1}.$$

The proof therefore follows from that of theorem 1.

References

d'Aspremont, A., El Ghaoui, L., Jordan, M. I. and Lanckriet, G. R. G. (2007) A direct formulation for sparse PCA using semidefinite programming. *SIAM Rev.*, **49**, 434–448.

Aston, J. A. D. and Kirch, C. (2012) Evaluating stationarity via change-point alternatives with applications to fMRI data. *Ann. Appl. Statist.*, **6**, 1906–1948.

Aston, J. A. D. and Kirch, C. (2014) Change points in high dimensional settings. *Preprint arXiv:1409.1771*. Statistical Laboratory. University of Cambridge, Cambridge.

Aue, A., Hörmann, S., Horváth, L. and Reimherr, M. (2009) Break detection in the covariance structure of multivariate time series models. *Ann. Statist.*, **37**, 4046–4087.

Bai, J. (2010) Common breaks in means and variances for panel data. *J. Econmetr.*, **157**, 78–92.

Bickel, P. J., Ritov, Y. and Tsybakov, A. B. (2009) Simultaneous analysis of Lasso and Dantzig selector. *Ann. Statist.*, **37**, 1705–1732.

Bleakley, K. and Vert, J. P. (2011) The group fused lasso for multiple change-point detection. *Technical Report HAL-00602121*. Computational Biology Center, Paris.

Boyd, S., Parikh, N., Chu, E., Peleato, B. and Eckstein, J. (2011) Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, **3**, 1–122.

Bücher, A., Kojadinovic, I., Rohmer, T. and Seger, J. (2014) Detecting changes in cross-sectional dependence in multivariate time series. *J. Multiv. Anal.*, **132**, 111–128.

Chen, J. and Gupta, A. K. (1997) Testing and locating variance changepoints with application to stock prices. *J. Am. Statist. Ass.*, **92**, 739–747.

Chen, Y. and Ye, X. (2011) Projection onto a simplex. *Preprint arXiv:1101.6081*. University of Florida, Gainesville.

Cho, H. (2016) Change-point detection in panel data via double CUSUM statistic. *Electron. J. Statist.*, **10**, 2000–2038.

Cho, H. and Fryzlewicz, P. (2015) Multiple-change-point detection for high dimensional time series via sparsified binary segmentation. *J. R. Statist. Soc.* B, **77**, 475–507.

Cribben, I. and Yu, Y. (2015) Estimating whole brain dynamics using spectral clustering. *Preprint arXiv: 1509.03730*. University of Cambridge, Cambridge.

Csörgő, M. and Horváth, L. (1997) *Limit Theorems in Change-point Analysis*. New York: Wiley.

Cule, M., Samworth, R. and Stewart, M. (2010) Maximum likelihood estimation of a multi-dimensional log-concave density (with discussion). *J. R. Statist. Soc.* B, **72**, 545–607.

Darling, D. A. and Erdős, P. (1956) A limit theorem for the maximum of normalised sums of independent random variables. *Duke Math. J.*, **23**, 143–155.

Davis, C. and Kahan, W. M. (1970) The rotation of eigenvectors by a pertubation: III. *SIAM J. Numer. Anal.*, **7**, 1–46.

Dümbgen, L. and Rufibach, K. (2009) Maximum likelihood estimation of a log-concave density and its distribution function: basic properties and uniform consistency. *Bernoulli*, **15**, 40–68.

Enikeeva, F. and Harchaoui, Z. (2014) High-dimensional change-point detection with sparse alternatives. *Preprint arXiv:1312.1900v2*. Laboratoire Jean Kuntzmann, Grenoble.

Frick, K., Munk, A. and Sieling, H. (2014) Multiscale change point inference (with discussion). *J. R. Statist. Soc.* B, **76**, 495–580.

Fryzlewicz, P. (2014) Wild binary segmentation for multiple change-point detection. *Ann. Statist.*, **42**, 2243–2281.

Gabay, D. and Mercier, B. (1976) A dual algorithm for the solution of nonlinear variational problems via finite element approximations. *Comput. Math. Appl.*, **2**, 17–40.

Hampel, F. R. (1974) The influence curve and its role in robust estimation. *J. Am. Statist. Ass.*, **69**, 383–393.

Henry, D., Simani, S. and Patton, R. J. (2010) Fault detection and diagnosis for aeronautic and aerospace missions. In *Fault Tolerant Flight Control—a Benchmark Challenge* (eds C. Edwards, T. Lombaerts and H. Smaili), pp. 91–128. Berlin: Springer.

Horváth, L. and Hušková, M. (2012) Change-point detection in panel data. *J. Time Ser. Anal.*, **33**, 631–648.

Horváth, L., Kokoszka, P. and Steinebach, J. (1999) Testing for changes in dependent observations with an application to temperature changes. *J. Multiv. Anal.*, **68**, 96–199.

Horváth, L. and Rice, G. (2014) Extensions of some classical methods in change point analysis. *Test*, **23**, 219–255.

Hubert, L. and Arabie, P. (1985) Comparing partitions. *J. Classificn*, **2**, 193–218.

James, N. A. and Matteson, D. S. (2015) ecp: an R package for nonparametric multiple change point analysis of multivariate data. *J. Statist. Softwr.*, **62**, 1–25.

Jirak, M. (2015) Uniform change point tests in high dimension. *Ann. Statist.*, **43**, 2451–2483.

Johnstone, I. M. and Lu, A. Y. (2009) On consistency and sparsity for principal components analysis in high dimensions. *J. Am. Statist. Ass.*, **104**, 682–693.

Killick, R., Fearnhead, P. and Eckley, I. A. (2012) Optimal detection of changepoints with a linear computational cost. *J. Am. Statist. Ass.*, **107**, 1590–1598.

Kirch, C., Mushal, B. and Ombao, H. (2015) Detection of changes in multivariate time series with applications to EEG data. *J. Am. Statist. Ass.*, **110**, 1197–1216.

Lavielle, M. and Teyssiere, G. (2006) Detection of multiple change-points in multivariate time series. *Lith. Math. J.*, **46**, 287–306.

Olshen, A. B., Venkatraman, E. S., Lucito, R. and Wigler, M. (2004) Circular binary segmentation for the analysis of array-based DNA copy number data. *Biometrika*, **5**, 557–572.

Ombao, H., Von Sachs, R. and Guo, W. (2005) SLEX analysis of multivariate nonstationary time series. *J. Am. Statist. Ass.*, **100**, 519–531.

Page, E. S. (1955) A test for a change in a parameter occurring at an unknown point. *Biometrika*, **42**, 523–527.

Peng, T., Leckie, C. and Ramamohanarao, K. (2004) Proactively detecting distributed denial of service attacks using source IP address monitoring. In *Networking 2004* (eds N. Mitrou, K. Kontovasilis, G. N. Rouskas, I. Iliadis and L. Merakos), pp. 771–782. Berlin: Springer.

Preuß, P., Puchstein, R. and Dette, H. (2015) Detection of multiple structural breaks in multivariate time series. *J. Am. Statist. Ass.*, **110**, 654–668.

Rand, W. M. (1971) Objective criteria for the evaluation of clustering methods. *J. Am. Statist. Ass.*, **66**, 846–850.

Soh, Y. S. and Chandrasekaran, V. (2017) High-dimensional change-point estimation: combining filtering with convex optimization. *Appl. Comp. Harm. Anal.*, **43**, 122–147.

Sparks, R., Keighley, T. and Muscatello, D. (2010) Early warning CUSUM plans for surveillance of negative binomial daily disease counts. *J. Appl. Statist.*, **37**, 1911–1930.

Tartakovsky, A., Nikiforov, I. and Basseville, M. (2014) *Sequential Analysis: Hypothesis Testing and Changepoint Detection*. Boca Raton: CRC Press.

Tillmann, A. N. and Pfetsch, M. E. (2014) The computational complexity of the restricted isometry property, the nullspace property, and related concepts in compressed sensing. *IEEE Trans. Inform. Theory*, **60**, 1248–1259.

Venkatraman, E. S. (1992) Consistency results in multiple change-point problems. *Doctoral Dissertation*. Department of Statistics, Stanford University, Stanford.

Vu, V. Q., Cho, J., Lei, J. and Rohe, K. (2013) Fantope projection and selection: a near-optimal convex relaxation of sparse PCA. *Adv. Neurl Inform. Process. Syst.*, **26**.

Wang, T., Berthet, Q. and Samworth, R. J. (2016) Statistical and computational trade-offs in estimation of sparse principal components. *Ann. Statist.*, **44**, 1896–1930.

Wang, T. and Samworth, R. J. (2016) InspectChangepoint: high-dimensional changepoint estimation via sparse projection. *R Package Version 1.0*. Statistical Laboratory, University of Cambridge, Cambridge. (Available from `https://cran.r-project.org/web/packages/InspectChangepoint/`.)

Yu, Y., Wang, T. and Samworth, R. J. (2015) A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika*, **102**, 315–323.

Zhang, N. R., Siegmund, D. O., Ji, H. and Li, J. Z. (2010) Detecting simultaneous changepoints in multiple sequences. *Biometrika*, **97**, 631–645.

Supporting information

Additional ‘supporting information’ may be found in the on-line version of this article:

‘High-dimensional changepoint estimation via sparse projection’.