High dimensional change point estimation via sparse projection

Tengyao Wang, Richard Samworth

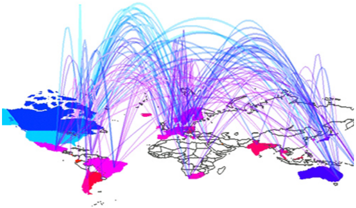
University College London, University of Cambridge

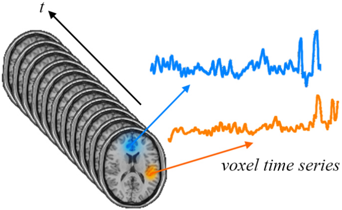
Royal Statistical Society Journal Webinar

22 Jan 2020

Motivation

- Many modern applications involve time-ordered high-dimensional data.





- Heterogeneity is a common feature of high-dimensional data, which is typically manifested through non-stationarity for data streams.
- Change point analysis can be used as a first step towards handling such heterogeneity.

Landscape of change point literature

- Change point analysis dates back at least to Page (1955).
- In the univariate setting, the state-of-the-art methods include PELT (Killick, Fearnhead and Eckley, 2012), WBS (Fryzlewicz, 2014) and SMUCE (Frick, Munk and Sieling, 2014).
- Some of the univariate change point methodologies have been extended to multivariate settings. (Horváth, Kokoszka and Steinebach, 1999; Ombao, Von Sachs and Guo, 2005; Aue et al., 2009; Kirch, Mushal and Ombao, 2014).
- Increasing interest in high-dimensional settings. (Aston and Kirch, 2014; Enikeeva and Harchaoui, 2014; Jirak, 2015; Cho and Fryzlewicz, 2015; Cho, 2016).
- We propose a new method, **inspect**, based on convex optimisation.

Basic model

- **Data generating mechanism:** $X = (X_1, \ldots, X_n) \in \mathbb{R}^{p \times n}$ with independent $X_1, \ldots, X_n$ such that for $1 \leq t \leq n$,

$$X_t \sim N_p(\mu_t, \sigma^2 I_p).$$

- Change points: $1 \leq z_1 < \cdots < z_\nu \leq n - 1$. ($z_0 := 0$ and $z_{\nu+1} := n$)
- Piecewise constant mean structure

$$\mu_{z_i+1} = \cdots = \mu_{z_{i+1}} \quad =: \ \mu^{(i)}, \quad 0 \leq i \leq \nu.$$

Vectors of change $\theta^{(i)} := \mu^{(i)} - \mu^{(i-1)}$.

Basic model

- **Additional assumptions**:

  Spatial sparsity of changes

$$\|\theta^{(i)}\|_0 \le k, \qquad \forall\, 1 \le i \le \nu.$$

  Minimal signal strength

$$\|\theta^{(i)}\|_2 \ge \vartheta, \qquad \forall\, 1 \le i \le \nu.$$

  Stationary run lengths satisfy

$$z_{i+1} - z_i \ge n\tau, \qquad \forall\, 0 \le i \le \nu.$$

Basic model

- **Additional assumptions**:

  Spatial sparsity of changes

$$\|\theta^{(i)}\|_0 \leq k, \qquad \forall\, 1 \leq i \leq \nu.$$

  Minimal signal strength

$$\|\theta^{(i)}\|_2 \geq \vartheta, \qquad \forall\, 1 \leq i \leq \nu.$$

  Stationary run lengths satisfy

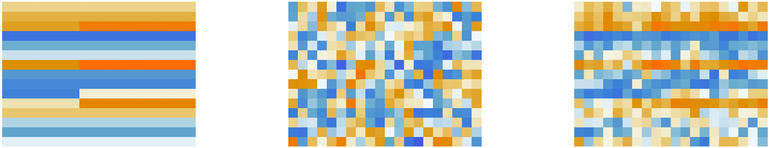
$$z_{i+1} - z_i \geq n\tau, \qquad \forall\, 0 \leq i \leq \nu.$$

- Let $\mathcal{P}(n, p, k, \nu, \vartheta, \tau, \sigma^2)$ be the set of distributions satisfying the above assumptions.

Estimating a single change point

Estimating a single change point

- Let $\nu = 1$, write $z := z_1$, $\theta := \theta^{(1)}$ and $\tau := n^{-1}\min\{z, n - z\}$.



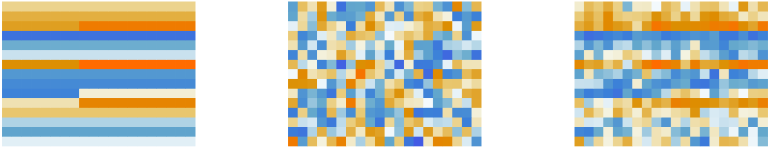
$$\boldsymbol{\mu} \quad + \quad W \quad = \quad X$$

- Optimal projection direction is $\theta / \|\theta\|_2 =: v$.
- (Sparse) principal component analysis? **Inefficient use of temporal information**.

Temporal aggregation

- Use CUSUM transformation $\mathcal{T} : \mathbb{R}^{p\times n} \to \mathbb{R}^{p\times(n-1)}$ for temporal aggregation:

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

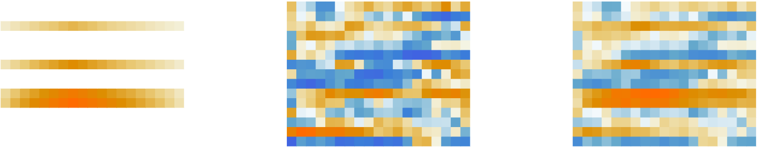


$$\boldsymbol{\mu} \quad + \quad W \quad = \quad X$$

Temporal aggregation

- Use CUSUM transformation $\mathcal{T} : \mathbb{R}^{p\times n} \to \mathbb{R}^{p\times(n-1)}$ for temporal aggregation:

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

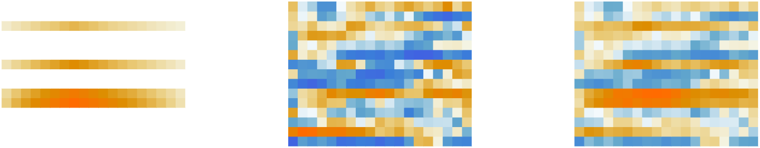


$$\mathcal{T}(\boldsymbol{\mu}) \quad + \quad \mathcal{T}(W) \quad = \quad \mathcal{T}(X)$$

Temporal aggregation

- Use CUSUM transformation $\mathcal{T} : \mathbb{R}^{p\times n} \to \mathbb{R}^{p\times(n-1)}$ for temporal aggregation:

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



$A \quad + \quad E \quad = \quad T$

Denote $A := \mathcal{T}(\boldsymbol{\mu})$, $E := \mathcal{T}(W)$ and $T := \mathcal{T}(X)$.

Estimating the oracle projection direction

- For a single change point, we can write $A := \mathcal{T}(\boldsymbol{\mu})$ explicitly as

$$A_{j,t} = \begin{cases} \sqrt{\frac{t}{n(n-t)}}(n-z)\theta_j, & \text{if } t \le z \\ \sqrt{\frac{n-t}{nt}} z\theta_j, & \text{if } t > z. \end{cases}$$

  Oracle projection direction v is the leading left singular vector of A.
- We could therefore estimate v by

$$\hat{v}_{\max,k} \in \operatorname*{arg\,max}_{u \in \mathbb{S}^{p-1}(k)} \|u^\top T\|_2.$$

Estimating the oracle projection direction

- For a single change point, we can write $A := \mathcal{T}(\boldsymbol{\mu})$ explicitly as

$$A_{j,t} = \begin{cases} \sqrt{\frac{t}{n(n-t)}}(n-z)\theta_j, & \text{if } t \le z \\ \sqrt{\frac{n-t}{nt}} z\theta_j, & \text{if } t > z. \end{cases}$$

  Oracle projection direction v is the leading left singular vector of A.

- We could therefore estimate v by

$$\hat{v}_{\max,k} \in \operatorname*{arg\,max}_{u \in \mathbb{S}^{p-1}(k)} \|u^\top T\|_2.$$

Proposition. When $n \ge 6$, with probability at least $1 - 4(p \log n)^{-1/2}$,

$$\sin \measuredangle(\hat{v}_{\max,k}, v) \le \frac{16\sqrt{2}\sigma}{\tau\vartheta}\sqrt{\frac{k \log(p \log n)}{n}}.$$

Estimating the oracle projection direction

- For a single change point, we can write $A := \mathcal{T}(\boldsymbol{\mu})$ explicitly as

$$A_{j,t} = \begin{cases} \sqrt{\frac{t}{n(n-t)}}(n-z)\theta_j, & \text{if } t \leq z \\ \sqrt{\frac{n-t}{nt}}z\theta_j, & \text{if } t > z. \end{cases}$$

  Oracle projection direction v is the leading left singular vector of A.
- We could therefore estimate v by

$$\hat{v}_{\max,k} \in \operatorname*{arg\,max}_{u \in \mathbb{S}^{p-1}(k)} \|u^\top T\|_2.$$

Proposition. When $n \geq 6$, with probability at least $1 - 4(p \log n)^{-1/2}$,

$$\sin \measuredangle(\hat{v}_{\max,k}, v) \leq \frac{16\sqrt{2}\sigma}{\tau\vartheta}\sqrt{\frac{k \log(p \log n)}{n}}.$$

But computing $\hat{v}_{\max,k}$ is **NP-hard**! (Tillmann and Pfetsch, 2014)

Convex relaxation

- How to make the non-convex problem convex? First, **lift** the original problem into a matrix optimisation problem.

$$\max_{u \in \mathbb{S}^{p-1}(k)} \|u^\top T\|_2 = \max_{u \in \mathbb{S}^{p-1}(k), w \in \mathbb{S}^{n-2}} u^\top T w$$
$$= \max_{u \in \mathbb{S}^{p-1}(k), w \in \mathbb{S}^{n-2}} \langle u w^\top, T \rangle = \max_{M \in \mathcal{M}} \langle M, T \rangle,$$

where $\mathcal{M} := \{M : \|M\|_* = 1, \mathrm{rk}(M) = 1, \mathrm{nnzr}(M) \leq k\}$. and $\langle A, B \rangle := \mathrm{tr}(A^\top B)$ is the trace inner product.

Convex relaxation

- How to make the non-convex problem convex? First, **lift** the original problem into a matrix optimisation problem.

$$\max_{u \in \mathbb{S}^{p-1}(k)} \|u^\top T\|_2 = \max_{u \in \mathbb{S}^{p-1}(k), w \in \mathbb{S}^{n-2}} u^\top T w$$
$$= \max_{u \in \mathbb{S}^{p-1}(k), w \in \mathbb{S}^{n-2}} \langle uw^\top, T \rangle = \max_{M \in \mathcal{M}} \langle M, T \rangle,$$

where $\mathcal{M} := \{M : \|M\|_* = 1, \mathrm{rk}(M) = 1, \mathrm{nnzr}(M) \leq k\}$. and $\langle A, B \rangle := \mathrm{tr}(A^\top B)$ is the trace inner product.

- Find a **convex relaxation** of the above matrix problem

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

where $\mathcal{S}_1 := \{M \in \mathbb{R}^{p \times (n-1)} : \|M\|_* \leq 1\}$.

Further relaxation

- The optimiser $\hat{M} \in \arg\max_{M \in \mathcal{S}_1} \{\langle M, T \rangle - \lambda \|M\|_1\}$ can be computed via alternating direction method of multipliers (ADMM).
- Polynomial time computable, but still slow for large datasets.

Further relaxation

- The optimiser $\hat{M} \in \arg\max_{M \in \mathcal{S}_1} \big\{ \langle M, T \rangle - \lambda \|M\|_1 \big\}$ can be computed via alternating direction method of multipliers (ADMM).
- Polynomial time computable, but still slow for large datasets.
- Further enlarge the admissible set to

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

Further relaxation

- The optimiser $\hat{M} \in \arg\max_{M \in \mathcal{S}_1} \big\{ \langle M, T \rangle - \lambda \|M\|_1 \big\}$ can be computed via alternating direction method of multipliers (ADMM).
- Polynomial time computable, but still slow for large datasets.
- Further enlarge the admissible set to

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

- Much easier to compute:

$$\hat{M} := \frac{\mathbf{soft}(T, \lambda)}{\|\mathbf{soft}(T, \lambda)\|_2} \in \arg\max_{M \in \mathcal{S}_2} \big\{ \langle T, M \rangle - \lambda \|M\|_1 \big\}.$$

- But the relaxed problem is further away from the original non-convex problem.

Further relaxation

- The optimiser $\hat{M} \in \arg\max_{M \in \mathcal{S}_1} \{\langle M, T\rangle - \lambda \|M\|_1\}$ can be computed via alternating direction method of multipliers (ADMM).
- Polynomial time computable, but still slow for large datasets.
- Further enlarge the admissible set to

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

- Much easier to compute:

$$\hat{M} := \frac{\mathbf{soft}(T, \lambda)}{\|\mathbf{soft}(T, \lambda)\|_2} \in \arg\max_{M \in \mathcal{S}_2} \{\langle T, M\rangle - \lambda \|M\|_1\}.$$

- But the relaxed problem is further away from the original non-convex problem. …**Statistical and computational trade-off**.

Estimating the oracle projection direction

In both relaxations ($\mathcal{S}_1$ or $\mathcal{S}_2$), we estimate the oracle projection direction by

$$\hat{v} := \text{leading left singular vector of } \hat{M}.$$

Proposition. If $n \geq 6$ and $\lambda = 2\sigma\sqrt{\log(p \log n)}$, then with probability at least $1 - 4(p \log n)^{-1/2}$,

$$\sin \measuredangle(\hat{v}, v) \leq \frac{64\sigma}{\tau\vartheta}\sqrt{\frac{k \log(p \log n)}{n}}.$$

Estimating a single change point

Algorithm for a single change point

Input $X \in \mathbb{R}^{p \times n}, \lambda > 0$
Step 1 CUSUM transformation $T \leftarrow \mathcal{T}(X)$
Step 2 Find

$$\hat{M} \in \arg\max_{M \in \mathcal{S}} \{\langle T, M \rangle - \lambda \|M\|_1\} \qquad (\mathcal{S} = \mathcal{S}_1 \text{ or } \mathcal{S}_2)$$

Step 3 Set $\hat{v} \leftarrow$ the leading left singular vector of $\hat{M}$
Step 4 Let $\bar{T} \leftarrow \hat{v}^\top T$, set $\hat{z} \leftarrow \arg\max_t |\bar{T}_t|$ and $\bar{T}_{\max} \leftarrow \max_t |\bar{T}_t|$.
Output $\hat{z}$ and $\bar{T}_{\max}$.

Theoretical guarantees

Theoretical performance of a sample-splitting version of the algorithm:

Theorem. Suppose σ is known and $X \sim \mathbf{P} \in \mathcal{P}(n, p, k, 1, \vartheta, \tau, \sigma^2)$. Let $\hat{z}$ be the output of the sample-splitting algorithm with input X, σ and $\lambda = 2\sigma\sqrt{\log(p \log n)}$. If $n \geq 12$ and $\frac{C\sigma}{\vartheta\tau}\sqrt{\frac{k\log(p\log n)}{n}} \leq 1$, then

$$\mathbf{P}\left(\frac{1}{n}|\hat{z} - z| \leq \frac{C'\sigma^2 \log\log n}{n\vartheta^2}\right) \geq 1 - \frac{7}{\sqrt{\log(n/2)}}$$

e.g. consider the setting: $\log p = O(\log n)$, $\vartheta \asymp n^{-a}$, $\tau \asymp n^{-b}$, $k \asymp n^c$. If $a + b + c/2 < 1/2$, then rate of convergence is $o(n^{-1+2a+\delta})$ for all $\delta > 0$.

Theoretical guarantees

Theoretical performance of a sample-splitting version of the algorithm:

Theorem. Suppose σ is known and $X \sim \mathbf{P} \in \mathcal{P}(n, p, k, 1, \vartheta, \tau, \sigma^2)$. Let $\hat{z}$ be the output of the sample-splitting algorithm with input X, σ and $\lambda = 2\sigma\sqrt{\log(p \log n)}$. If $n \geq 12$ and $\frac{C\sigma}{\vartheta\tau}\sqrt{\frac{k\log(p\log n)}{n}} \leq 1$, then

$$\mathbf{P}\left(\frac{1}{n}|\hat{z} - z| \leq \frac{C'\sigma^2 \log\log n}{n\vartheta^2}\right) \geq 1 - \frac{7}{\sqrt{\log(n/2)}}$$

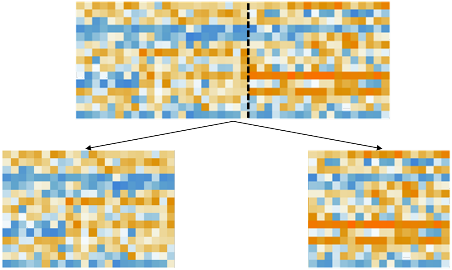
e.g. consider the setting: $\log p = O(\log n)$, $\vartheta \asymp n^{-a}$, $\tau \asymp n^{-b}$, $k \asymp n^c$. If $a + b + c/2 < 1/2$, then rate of convergence is $o(n^{-1+2a+\delta})$ for all $\delta > 0$.

Minimax optimal up to $\log\log n$.

Estimating multiple change points

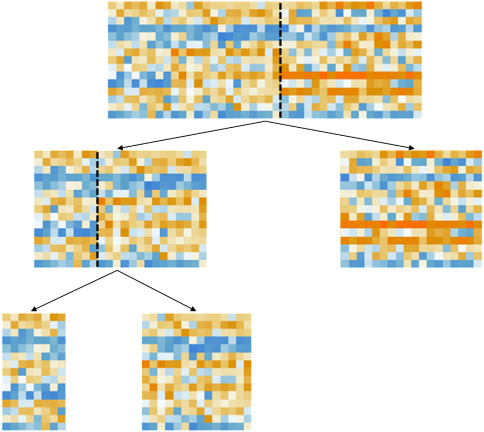
Binary segmentation

A top-down approach:



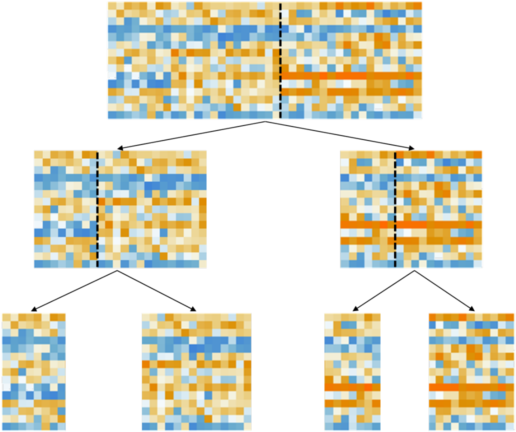
Binary segmentation

A top-down approach:



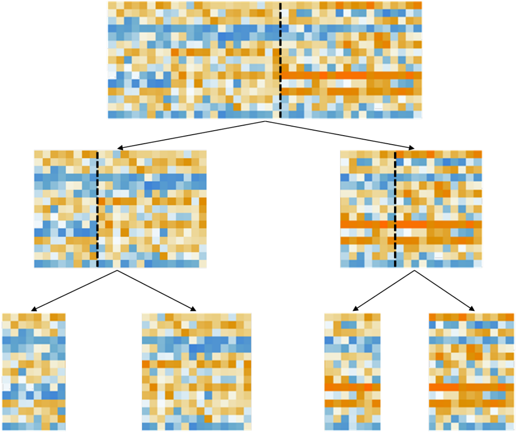
Binary segmentation

A top-down approach:



Binary segmentation

A top-down approach:

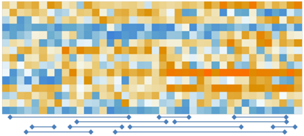


But…multiple change points may offset each other.

Misaligned change coordinates result in bad projection direction estimator.

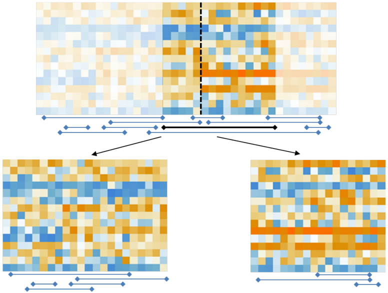
Wild binary segmentation

Wild binary segmentation scheme (Fryzlewicz, 2014)



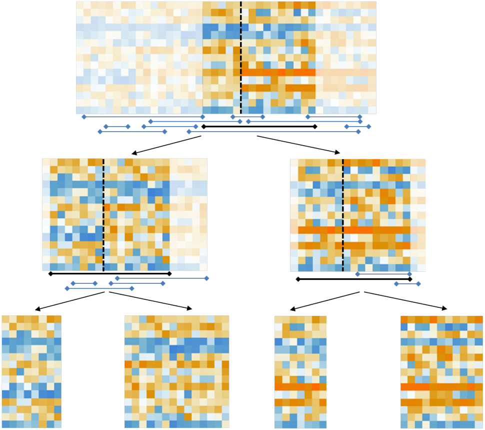
Wild binary segmentation

Wild binary segmentation scheme (Fryzlewicz, 2014)



Wild binary segmentation

Wild binary segmentation scheme (Fryzlewicz, 2014)



Estimating multiple change points

Algorithm **inspect** ('InspectChangepoint' R package)

Input $X \in \mathbb{R}^{p \times n}, \lambda \geq 0, \xi \geq 0, \beta \geq 0, Q \in \mathbb{N}$

Step 1 Sample Q random intervals $[s_1, e_1], \ldots, [s_Q, e_Q]$

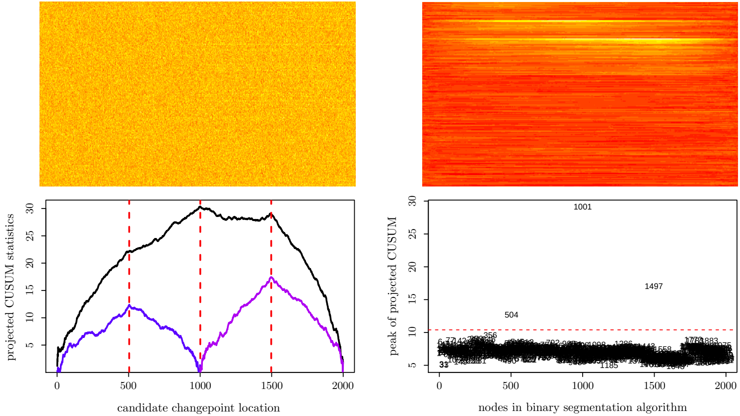
Step 2 Run **wbs**$(0, n)$, where function **wbs**(s, e) is defined by

- $\mathcal{Q}_{s,e} \leftarrow \{q : [s_q, e_q] \subseteq [s + n\beta, e - n\beta]\}$.
- For each $q \in \mathcal{Q}_{s,e}$, run single change point algorithm with input $X^{[s_q, e_q]}$ and λ to obtain $\hat{z}^{[q]}$ and $\bar{T}_{\max}^{[q]}$.
- Find $q_0 \in \arg\max_q \bar{T}_{\max}^{[q]}$ and set $b \leftarrow s_{q_0} + \hat{z}^{[q_0]}$.
- If $\bar{T}_{\max}^{[q_0]} > \xi$, then add b to the set of estimated change points, and run **wbs**(s, b) and **wbs**(b, e).

Output ordered estimated change points $\hat{z}_1, \ldots, \hat{z}_{\hat{\nu}}$.

Estimating multiple change points

An example: **inspect** in action



Theoretical guarantees

Theoretical performance of a sample-splitting version of **inspect**:

Theorem. Suppose σ is known and $X \sim \mathbf{P} \in \mathcal{P}(2n, p, k, 1, \vartheta, \tau, \sigma^2)$. Let $\hat{z}_1 < \cdots < \hat{z}_{\hat{\nu}}$ be the output of the sample-splitting algorithm with input X, σ, $\lambda := 4\sigma\sqrt{\log(np)}$, $\xi := \lambda$, β and Q. Define $\rho = \rho_n := \lambda^2 n^{-1}\vartheta^{-2}\tau^{-5}$. If $n\tau \geq 14$, $2\rho < \beta < \frac{2}{9}\tau$ and $C\rho k\tau^3 \leq 1$, then

$$\mathbb{P}_{\mathbf{P}}\left\{\hat{\nu} = \nu \text{ and } \frac{1}{n}|\hat{z}_i - z_i| \leq C'\rho \quad \forall\, i\right\} \geq 1 - \frac{e^{-\tau^2 Q/9}}{\tau} - \frac{3\log n}{np^4}.$$

e.g. consider the setting $\log p = O(\log n)$, $\vartheta \asymp n^{-a}$, $\tau \asymp n^{-b}$, $k \asymp n^c$. If $a + b + c/2 < 1/2$ and $a + 3b < 1/2$, then $\rho_n = o(n^{-1+2a+5b+\delta})$ for all $\delta > 0$.

Numerical studies

Numerical studies

We compare **inspect** algorithm with other recently proposed methods

- Sparsified Binary Segmentation (**sbs**) (Cho and Fryzlewicz, 2015)
- the Double CUSUM algorithm (**dc**) (Cho, 2016)
- a scan statistic-based algorithm (**scan**) (Enikeeva and Harchaoui, 2014)
- an ℓ_∞ CUSUM aggregation algorithm ($\mathbf{agg}_\infty$) (Jirak, 2015)
- an ℓ_2 CUSUM aggregation algorithm ($\mathbf{agg}_2$) (Horvath and Huskova, 2012)

Single change point estimation

n	p	k	z	ϑ	**inspect**	**dc**	**sbs**	**scan**	**agg$_2$**	**agg$_\infty$**
1000	1000	3	400	0.8	9.5	14.6	117.2	**9.0**	154.9	15.0
1000	1000	32	400	0.8	**20.7**	61.1	83.6	26.4	150.1	57.2
1000	1000	100	400	0.8	**33.1**	101.0	122.0	59.2	158.3	106.4
1000	1000	1000	400	0.8	**57.7**	159.9	186.3	145.2	152.7	195.2
1000	2000	3	400	0.8	10.8	15.4	132.9	**10.3**	232.8	15.5
1000	2000	45	400	0.8	**29.6**	121.0	137.0	39.1	237.5	73.4
1000	2000	200	400	0.8	**47.4**	176.8	187.7	123.6	235.4	158.2
1000	2000	2000	400	0.8	**67.2**	219.6	240.0	210.3	233.4	245.8
2000	1000	3	800	0.8	**8.1**	14.2	178.3	8.3	42.6	14.4
2000	1000	32	800	0.8	**12.5**	36.1	58.7	16.9	40.6	38.2
2000	1000	100	800	0.8	**17.0**	46.7	75.8	24.6	40.0	47.3
2000	1000	1000	800	0.8	**31.0**	89.0	111.2	45.4	39.9	91.0
2000	2000	3	800	0.8	9.3	15.9	215.7	**9.0**	143.6	16.1
2000	2000	45	800	0.8	**16.7**	35.8	100.7	21.3	152.5	39.2
2000	2000	200	800	0.8	**25.6**	56.7	126.5	32.0	151.8	59.1
2000	2000	2000	800	0.8	**48.4**	107.9	208.0	66.1	150.6	153.5

Table: Root mean squared error in single change point estimation by different algorithms.

Single change point estimation

Distribution of estimated change point location

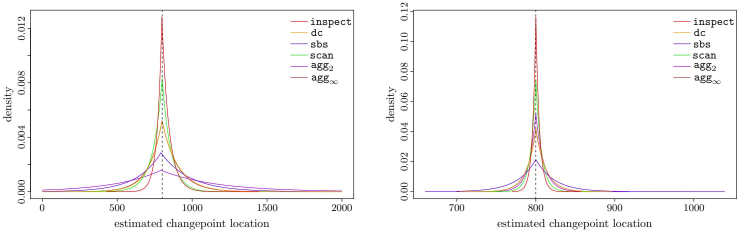


Figure: Estimated densities of location of change point estimates by different algorithms. Left panel: $(n, p, k, z, \vartheta, \sigma^2) = (2000, 1000, 32, 800, 0.5, 1)$; right panel: $(n, p, k, z, \vartheta, \sigma^2) = (2000, 1000, 32, 800, 1, 1)$.

Multiple change points estimation

Three change points at $500, 1000, 1500$. Writing $\vartheta^{(i)} := \|\theta^{(i)}\|_2/\|\theta^{(i)}\|_0^{1/2}$, set $(\vartheta^{(1)}, \vartheta^{(2)}, \vartheta^{(3)}) = (\vartheta, 2\vartheta, 3\vartheta)$.

$(\vartheta^{(1)}, \vartheta^{(2)}, \vartheta^{(3)})$	method	$\hat{\nu}$						ARI	% best
		0	1	2	3	4	5		
$(0.6, 1.2, 1.8)$	**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$_\infty$	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$_\infty$	0	0	82	17	1	0	0.69	12

Table: Multiple change point simulation results. Other simulation parameters: $n = 2000$, $p = 200$, $k = 40$, $z = (500, 1000, 1500)$ and $\sigma^2 = 1$.

Multiple change point estimation

Distribution of estimated chang-epoint locations

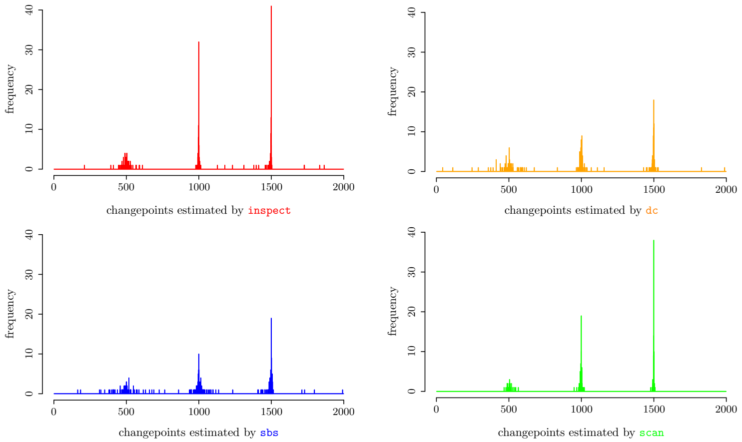


Figure: Histograms of estimated change point locations by **inspect**, **dc**, **sbs** and **scan**. Parameters: $n = 2000$, $p = 200$, $k = 40$, $z = (500, 1000, 1500)$, $(\vartheta^{(1)}, \vartheta^{(2)}, \vartheta^{(3)}) = (0.6, 1.2, 1.8)$, $\sigma^2 = 1$.

Model misspecification

Robustness of our algorithm:

Model	**inspect**	**dc**	**sbs**	**scan**	**agg$_2$**	**agg$_\infty$**
$\mathrm{M}_{\mathrm{unif}}$	**2.7**	9.6	17.1	4.9	4.3	10.2
$\mathrm{M}_{\mathrm{exp}}$	**2.6**	9.6	42.6	5.0	4.7	9.6
$\mathrm{M}_{\mathrm{cs,loc}}(0.2)$	**3.5**	9.7	19.2	7.0	5.4	9.8
$\mathrm{M}_{\mathrm{cs,loc}}(0.5)$	**5.8**	9.7	24.6	8.7	9.3	9.6
$\mathrm{M}_{\mathrm{cs}}(0.5)$	**1.5**	7.7	14.9	3.0	3.6	6.7
$\mathrm{M}_{\mathrm{cs}}(0.9)$	**2.7**	9.9	18.6	4.7	4.7	9.6
$\mathrm{M}_{\mathrm{temp}}(0.1)$	**6.1**	20.3	102.8	9.4	10.9	20.2
$\mathrm{M}_{\mathrm{temp}}(0.3)$	**30.1**	32.4	276.4	38.8	38.2	34.8
$\mathrm{M}_{\mathrm{async}}(10)$	**5.8**	11.5	18.5	7.8	7.0	11.3

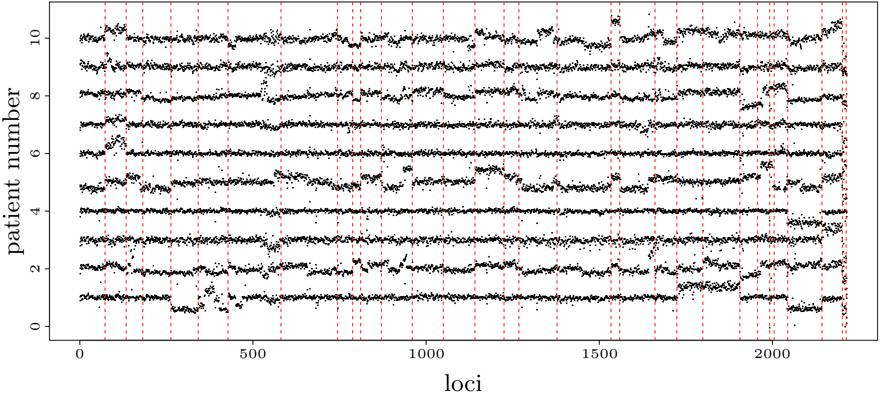
Table: Root mean squared error for different algorithms in single change point estimation, under different forms of model misspecification. Simulation parameters: $n = 2000$, $p = 1000$, $k = 32$, $z = 800$, $\vartheta = 1.5$.

Real data application

Copy number variation abnormality detection

Microarray dataset: 43 bladder cancer patients and 2215 loci.

Shared copy number abnormality regions likely disease related.



Summary

In this work

- We propose a new method for high-dimensional change point estimation.
- The key idea is optimal spatial aggregation via convex optimisation.
- **R** package **InspectChangepoint** is available on CRAN.

Summary

In this work

- We propose a new method for high-dimensional change point estimation.
- The key idea is optimal spatial aggregation via convex optimisation.
- **R** package **InspectChangepoint** is available on CRAN.

Future directions

- Handling non-trivial unknown spatial correlation
- Incorporating additional spatial structure in the signal
- Online method for high-dimensional change point esitmation

Summary

In this work

- We propose a new method for high-dimensional change point estimation.
- The key idea is optimal spatial aggregation via convex optimisation.
- **R** package **InspectChangepoint** is available on CRAN.

Future directions

- Handling non-trivial unknown spatial correlation
- Incorporating additional spatial structure in the signal
- Online method for high-dimensional change point esitmation

Thank you!