High-dimensional, multiscale online changepoint detection

Tengyao Wang
University College London

Joint Statistics Seminar, Fudan University
18 June 2020
Collaborators

Yudong Chen

Richard Samworth
Modern technology has facilitated the real-time monitoring of many types of evolving processes.
Modern technology has facilitated the real-time monitoring of many types of evolving processes.

Very often, a key feature of interest for data streams is a changepoint.
The vast majority of the literature concerns the offline problem (Killick et al., 2012; W. and Samworth, 2018; Wang et al., 2018; Baranowski et al., 2019; Liu et al., 2019).

Univariate online changepoints have been studied within the well-established field of statistical process control (Duncan, 1952; Page, 1954; Barnard, 1959; Fearnhead and Liu, 2007; Oakland, 2007).

Much less work on multivariate, online changepoint problems (Tartakovsky et al., 2006; Mei, 2010; Zou et al., 2015). Several methods involve scanning a moving window of fixed size (Xie and Siegmund, 2013; Soh and Chandrasekaran, 2017; Chan, 2017).
Key definition of an **online algorithm** for a data stream:

**Definition.** An algorithm is online if both its storage requirements and the computational complexity for processing a new observation depend only on the number of bits needed to represent the new data.
Online algorithm

Key definition of an **online algorithm** for a data stream:

**Definition.** An algorithm is online if both its storage requirements and the computational complexity for processing a new observation depend only on the **number of bits needed to represent the new data**.

- For the purposes of this definition, all real numbers are considered as floating point numbers.
- Importantly, we can only track a finite amount of summary statistics and are not allowed to store all historical data.
Problem setting

We consider a high-dimensional online changepoint detection problem:

- **Data**: for some unknown, deterministic time $z \in \mathbb{N} \cup \{0\}$, we have
  \[ X_1, \ldots, X_z \sim N_p(0, I_p) \text{ and } X_{z+1}, X_{z+2}, \ldots \sim N_p(\theta, I_p). \]

- $\theta = 0$: data generated under the null, i.e. no change.
- $\theta \neq 0$: data generated under the alternative, i.e. there exists a change.

- Assume $\vartheta := \|\theta\|_2$ is at least a known lower bound $\beta > 0$. 
A **sequential changepoint procedure** is an extended stopping time $N$ (w.r.t. the natural filtration) taking values in $\mathbb{N} \cup \{\infty\}$.

- **patience:** $E_0(N)$;

- Two types of **response delays**:
  - Average case response delay
    \[ \bar{E}_\theta(N) := \sup_{z \in \mathbb{N}} E_{z,\theta} \{(N - z) \lor 0\}; \]
  - Worst case response delay
    \[ \bar{E}^{wc}_\theta(N) := \sup_{z \in \mathbb{N}} \text{ess sup} E_{z,\theta} \{(N - z) \lor 0 \mid X_1, \ldots, X_z\}. \]
A warm-up: univariate online changepoint detection
Example of an online algorithm

Let $p = 1$ and assume $\theta > 0$. Page’s procedure (Page, 1954):

$$R_n := \max_{0 \leq h \leq n} \sum_{i=n-h+1}^{n} \beta(X_i - \beta/2) = \max \{ R_{n-1} + \beta(X_n - \beta/2), 0 \}.$$ 

Threshold $T \equiv T_\beta$ for changepoint declaration.
Example of an online algorithm

Let $p = 1$ and assume $\theta > 0$. Page’s procedure (Page, 1954):

$$R_n := \max_{0 \leq h \leq n} \sum_{i=n-h+1}^{n} \beta(X_i - \beta/2) = \max\{R_{n-1} + \beta(X_n - \beta/2), 0\}.$$ 

Threshold $T \equiv T_\beta$ for changepoint declaration.
Example of an online algorithm

Let $p = 1$ and assume $\theta > 0$. Page’s procedure (Page, 1954):

$$R_n := \max_{0 \leq h \leq n} \sum_{i=n-h+1}^{n} \beta(X_i - \beta/2) = \max\{R_{n-1} + \beta(X_n - \beta/2), 0\}.$$ 

Threshold $T \equiv T_\beta$ for changepoint declaration.

Page’s procedure has asymptotically optimal worst case response delay under a patience constraint (Lorden, 1971).
Example of an online algorithm?

Let $p = 1$ and assume $\theta > 0$. Scanning window-based method with window width $w > 0$:

$$W_n := \sum_{i=n-w+1}^{n} \beta(X_i - \beta/2).$$

- Window size $w$ needs to increase when $\beta$ decreases.
- Storage requirement depends on $\beta$. 

![Graph of scanning window statistic and cumulative log-likelihood with log-likelihood per observation](image)
Example of a non-online algorithm

Let $p = 1$ and assume $\theta > 0$. Shiryaev–Roberts procedure (Shiryaev, 1963; Roberts, 1966):

$$SR_n := \sum_{i=1}^{n} \prod_{h=i}^{n} e^{b(X_h - b/2)}.$$

- The statistics cannot be defined recursively
- A sequential but not online algorithm
A high-dimensional, multiscale online algorithm: ocd
Curse of dimensionality

- Page’s procedure in 1-d relies on the well-ordering of $\mathbb{R}$.
- Generalising to high dimensions:

![](image)

$n = 500$
Curse of dimensionality

- Page’s procedure in 1-d relies on the well-ordering of $\mathbb{R}$.
- Generalising to high dimensions:

  - If we know the direction of $\theta$, Page’s procedure can still be used.
  - Infeasible to examine all possible directions for a change for large $p$. 
Diagonal statistics

- Write $X_i = (X_{i1}, \ldots, X_{ip})^\top \in \mathbb{R}^p$. Fix $n \in \mathbb{N}$ and $b \in \mathbb{R}\setminus\{0\}$. For each $j \in [p]$, define (we have suppressed $n$ and $b$ dependence)

$$R_j := \max_{0 \leq h \leq n} \sum_{i=n-h+1}^{n} b(X_{ji} - b/2)$$

$$t_j := \arg\max_{0 \leq h \leq n} \sum_{i=n-h+1}^{n} b(X_{ji} - b/2).$$

- $(R_j)_{j \in [p]}$ are called the diagonal statistics.
For each $j$, compute normalised tail partial sums of length $t^j$ in all coordinates $j' \in [p]$:

$$A_{j',j} := \frac{1}{\sqrt{t^j}} \sum_{i=n-t^j+1}^{n} X_{i}^{j'} \sim_{\text{null}} N(0, 1).$$
For each $j$, compute normalised tail partial sums of length $t^j$ in all coordinates $j' \in [p]$:

$$A_{j',j} := \frac{1}{\sqrt{t^j}} \sum_{i=n-t^j+1}^{n} X_{i}^{j'} \sim_{\text{null}} N(0, 1).$$
For each $j$, compute normalised tail partial sums of length $t^j$ in all coordinates $j' \in [p]$:

$$A_{j', j} := \frac{1}{\sqrt{t^j}} \sum_{i=n-t^j+1}^{n} X_{i}^{j'} \sim_{\text{null}} N(0, 1).$$
For each $j$, compute normalised tail partial sums of length $t^j$ in all coordinates $j' \in [p]$:

$$A_{j',j} := \frac{1}{\sqrt{t^j}} \sum_{i=n-t^j+1}^{n} X_{i}^{j'} \sim_{\text{null}} N(0, 1).$$
For each $j$, compute normalised tail partial sums of length $t^j$ in all coordinates $j' \in [p]$:

$$A_{j',j} := \frac{1}{\sqrt{t^j}} \sum_{i=n-t^j+1}^{n} X_{i}^{j'} \sim_{\text{null}} N(0, 1).$$

We aggregate to form an off-diagonal statistic anchored at coordinate $j$:

$$Q^j := \sum_{j': j' \neq j} \left( A_{j',j} \right)^2 \sim_{\text{null}} \chi^2_{p-1}.$$
For each $j$, compute normalised tail partial sums of length $t^j$ in all coordinates $j' \in [p]$:

\[
A_{j',j} := \frac{1}{\sqrt{t^j}} \sum_{i=n-t^j+1}^n X_{i,j'}^{j'} \sim_{\text{null}} N(0, 1).
\]

We aggregate to form an off-diagonal statistic anchored at coordinate $j$:

\[
Q^j := \sum_{j' : j' \neq j} \left( A_{j',j} \mathbb{1}_{\{|A_{j',j}| \geq a\}} \right)^2 \quad \text{for some } a > 0.
\]
Vary the **scale** parameter $b$ over a (signed) dyadic grid

$$
\mathcal{B} := \left\{ \pm \frac{\beta}{\sqrt{2^\ell \log_2(2p)}} : \ell = 0, \ldots, \lfloor \log_2(2p) \rfloor \right\}.
$$

Aggregate diagonal and off-diagonal statistics from different coordinates and at different scales (recall $R^j_n$ and $Q^j_n$ both have $n$ and $b$ dependence):

$$
S_n^{\text{diag}} := \max_{(j,b) \in [p] \times \mathcal{B}} R^j_{n,b},
$$

$$
S_n^{\text{off}} := \max_{(j,b) \in [p] \times \mathcal{B}} Q^j_{n,b}.
$$

Declare change when either $S_n^{\text{diag}}$ or $S_n^{\text{off}}$ is large.
Algorithm 1: Pseudo-code of the ocd algorithm

Input: $X_1, X_2, \ldots \in \mathbb{R}^p$ observed sequentially, $\beta > 0$, $a \geq 0$, $T_{\text{diag}} > 0$ and $T_{\text{off}} > 0$

Set: $\mathcal{B} = \left\{ \pm \frac{\beta}{\sqrt{2^{\ell} \log_2 (2p)}} : \ell = 0, \ldots, \lfloor \log_2 p \rfloor \right\}$, $\mathcal{B}_0 = \left\{ \pm \frac{\beta}{\sqrt{2^{\lfloor \log_2 p \rfloor + 1} \log_2 (2p)}} \right\}$, $n = 0$, $A_b = 0 \in \mathbb{R}^{p \times p}$ and $t_b = 0 \in \mathbb{R}^{p}$ for all $b \in \mathcal{B} \cup \mathcal{B}_0$

repeat

$n \leftarrow n + 1$

observe new data vector $X_n$

for $(j, b) \in [p] \times (\mathcal{B} \cup \mathcal{B}_0)$ do

\[ t_b^j \leftarrow t_b^j + 1 \]

\[ A_b^{j,j} \leftarrow A_b^{j,j} + X_n \]

if $bA_b^{j,j} - b^2 t_b^j / 2 \leq 0$ then

\[ t_b^j \leftarrow 0 \text{ and } A_b^{j,j} \leftarrow 0 \]

compute $Q_b^j \leftarrow \sum_{j' \in [p]: j' \neq j} \frac{(A_b^{j',j})^2}{t_b^j + 1} \mathbb{I}\{|A_b^{j',j}| \geq a \sqrt{t_b^j} \}$

\[ S_{\text{diag}}^{\text{diag}} \leftarrow \max_{(j, b) \in [p] \times (\mathcal{B} \cup \mathcal{B}_0)} \left( bA_b^{j,j} - b^2 t_b^j / 2 \right) \]

\[ S_{\text{off}}^{\text{off}} \leftarrow \max_{(j, b) \in [p] \times \mathcal{B}} Q_b^j \]

until $S_{\text{diag}}^{\text{diag}} \geq T_{\text{diag}}$ or $S_{\text{off}}^{\text{off}} \geq T_{\text{off}}$

Output: $N = n$
Algorithm 1: Pseudo-code of the ocd algorithm

Input: $X_1, X_2 \ldots \in \mathbb{R}^p$ observed sequentially, $\beta > 0$, $a \geq 0$, $T^{\text{diag}} > 0$ and $T^{\text{off}} > 0$

Set: $\mathcal{B} = \left\{ \pm \frac{\beta}{\sqrt{2^\ell \log_2(2p)}} : \ell = 0, \ldots, \lfloor \log_2 p \rfloor \right\}$, $\mathcal{B}_0 = \left\{ \pm \frac{\beta}{\sqrt{2^{\lfloor \log_2 p \rfloor + 1} \log_2(2p)}} \right\}$, $n = 0, A_b = 0 \in \mathbb{R}^{p \times p}$ and $t_b = 0 \in \mathbb{R}^p$ for all $b \in \mathcal{B} \cup \mathcal{B}_0$

repeat

$n \leftarrow n + 1$

observe new data vector $X_n$

for $(j, b) \in [p] \times (\mathcal{B} \cup \mathcal{B}_0)$ do

$t_b^j \leftarrow t_b^j + 1$

$A_b^{j,j} \leftarrow A_b^{j,j} + X_n$

if $bA_b^{j,j} - b^2 t_b^j / 2 \leq 0$ then

$t_b^j \leftarrow 0$ and $A_b^{j,j} \leftarrow 0$

compute $Q_b^j \leftarrow \sum_{j' \in [p]: j' \neq j} \frac{(A_b^{j',j})^2}{t_b^{j'\downarrow} \sqrt{1}} \mathbb{1}\{ |A_b^{j',j}| \geq a \sqrt{t_b^{j'}} \}$

$S^{\text{diag}} \leftarrow \max_{(j, b) \in [p] \times (\mathcal{B} \cup \mathcal{B}_0)} (bA_b^{j,j} - b^2 t_b^j / 2)$

$S^{\text{off}} \leftarrow \max_{(j, b) \in [p] \times \mathcal{B}_0} Q_b^j$

until $S^{\text{diag}} \geq T^{\text{diag}}$ or $S^{\text{off}} \geq T^{\text{off}}$

Output: $N = n$
Dense, sparse and adaptive versions

- Choose $a = 0$ to detect a **dense** change.
- Choose $a = \sqrt{8 \log p}$ to detect a **sparse** change.

- The **adaptive** version runs two ocd algorithms with $a = 0$ and $a = \sqrt{8 \log p}$ in parallel and declares when either detects a change:

$$N := \inf \left\{ n : \frac{S_n}{T_{\text{diag}}} \lor \frac{S_n}{T_{\text{off,d}}} \lor \frac{S_n}{T_{\text{off,s}}} \geq 1 \right\}.$$
Setting: \( p = 100, z = 900, \vartheta = \beta = 1 \) and \( \gamma = 5000 \)
Theoretical analysis
Why does ocd work?

- **Patience**: guaranteed by choosing thresholds appropriately.

- **Response delay**:  
  - $S_{\text{diag}}$ detect changes that are concentrated in a single coordinate.  
  - $S_{\text{off}}$ aggregates signal across many coordinates.  
  - If the tail partial sum consists of post-change data only, then

$$Q^j := \sum_{j': j' \neq j} (A^{j',j})^2 \sim_{\text{altern.}} \chi_{p-1}^2 (t^j \| \theta^{-j} \|_2^2).$$
What if The last $t^j$ points contain some pre-change data?

We would like to aggregate over $\approx \frac{t^j}{2}$ points, so that $Q^j$ is eventually formed using post-change data only.

How can we achieve this in an online manner?
What if The last $t^j$ points contain some pre-change data?

We would like to aggregate over $\approx t^j/2$ points, so that $Q^j$ is eventually formed using post-change data only.
What if the last $t^j$ points contain some pre-change data?

We would like to aggregate over $\approx t^j/2$ points, so that $Q^j$ is eventually formed using post-change data only.

How can we achieve this in an online manner?
A toy problem

Given $X_1, X_2, \ldots \in \mathbb{R}$, how can we keep track of the sum of the final $\tau \approx t/2$ observations at time $t$ in an online way?
A toy problem

Given $X_1, X_2, \ldots \in \mathbb{R}$, how can we keep track of the sum of the final $\tau \approx t/2$ observations at time $t$ in an online way?

<table>
<thead>
<tr>
<th>$t$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>…</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau$</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>…</td>
</tr>
<tr>
<td>$\Lambda$</td>
<td>$X_1$</td>
<td>$X_2$</td>
<td>$X_2 + X_3$</td>
<td>$X_3 + X_4$</td>
<td>$X_3 + X_4 + X_5$</td>
<td>$X_3 + \cdots + X_6$</td>
<td>$X_3 + \cdots + X_7$</td>
<td>$X_5 + \cdots + X_8$</td>
<td>…</td>
</tr>
<tr>
<td>$\tilde{\Lambda}$</td>
<td>0</td>
<td>0</td>
<td>$X_3$</td>
<td>0</td>
<td>$X_5$</td>
<td>$X_5 + X_6$</td>
<td>$X_5 + X_6 + X_7$</td>
<td>0</td>
<td>…</td>
</tr>
</tbody>
</table>

$t/2 \leq \tau < 3t/4$ for $t \geq 2$. 
A slight variant: ocd'

Part of the modified algorithm, ocd', using ‘halved’ tail lengths:

\[
\text{for } (j, b) \in [p] \times (\mathcal{B} \cup \mathcal{B}_0) \text{ do}
\]

\[
t^j_b \leftarrow t^j_b + 1 \quad \text{and} \quad A^*_{b}^j \leftarrow A^*_{b}^j + X_n
\]

set \( \delta = 0 \) if \( t^j_b \) is a power of 2 and \( \delta = 1 \) otherwise.

\[
\begin{align*}
\tau^j_b & \leftarrow \tau^j_b \delta + \tilde{\tau}^j_b (1 - \delta) + 1 \\
\Lambda^*_{b}^j & \leftarrow \Lambda^*_{b}^j \delta + \tilde{\Lambda}^*_{b}^j (1 - \delta) + X_n \\
\tilde{\tau}^j_b & \leftarrow (\tilde{\tau}^j_b + 1) \delta \quad \text{and} \quad \tilde{\Lambda}^*_{b}^j \leftarrow (\tilde{\Lambda}^*_{b}^j + X_n) \delta.
\end{align*}
\]

\[
\text{if } bA_b^j - b^2 t^j_b / 2 \leq 0 \text{ then}
\]

\[
\begin{align*}
t^j_b & \leftarrow \tau^j_b \leftarrow \tilde{\tau}^j_b \leftarrow 0 \\
A^*_{b}^j & \leftarrow \Lambda^*_{b}^j \leftarrow \tilde{\Lambda}^*_{b}^j \leftarrow 0
\end{align*}
\]

compute \( Q^j_b \leftarrow \sum_{j' \in [p]: j' \neq j} \frac{(\Lambda^j_{b}^{j'})^2}{\tau^j_b \vee 1} 1_{\{\Lambda^j_{b}^{j'} \geq a \sqrt{\tau^j_b} \}} \)
Theoretical guarantees: patience

Choose thresholds

\[
\begin{align*}
T_{\text{diag}} &= \log \{24p\gamma \log_2(4p)\} \\
T_{\text{off},d} &= \psi \left(2 \log \{24p\gamma \log_2(2p)\} \right) \\
T_{\text{off},s} &= 8 \log \{24p\gamma \log_2(2p)\}
\end{align*}
\]

where \( \psi(x) = p - 1 + x + \sqrt{2(p - 1)x} \) and \( \gamma \geq 1 \) is a user-specified desired patience level.

**Theorem.** Assume there is no change. Then, the adaptive version of ocd' with the above choice of thresholds satisfies \( \mathbb{E}_0(N) \geq \gamma \).
Theoretical guarantees: response delay

The **effective sparsity** of $\theta \in \mathbb{R}^p$ is

$$s \equiv s(\theta) := \left| \left\{ j \in [p] : |\theta^j| \geq \frac{\|\theta\|_2}{\sqrt{s(\theta) \log_2(2p)}} \right\} \right|.$$  

**Theorem.** Assume that the post-change signal $\theta$ satisfies $\|\theta\|_2 = \vartheta \geq \beta > 0$ with effective sparsity $s$. Then, the adaptive version of ocd’ with the same choice of thresholds satisfies:

(a) (Worst case response delay)

$$\bar{E}^{wc}_\theta(N) \lesssim \frac{s \log(ep\gamma) \log(ep)}{\beta^2} \lor 1;$$

(b) (Average case response delay)

$$\bar{E}^{\theta}_\theta(N) \lesssim \left( \frac{\sqrt{p} \log(ep\gamma)}{\vartheta^2} \lor \frac{\sqrt{s \log(ep/\beta) \log(ep)}}{\beta^2} \right) \land \frac{s \log(ep\gamma) \log(ep)}{\beta^2},$$

for all sufficiently small $\beta < \beta_0(s)$. 

Online changepoint detection 26/33
Response delays vs. sparsity

Assume that $\vartheta \simeq \beta \lesssim 1$ and $\log(\gamma / \beta) \lesssim \log p$. Then

$$\bar{\mathbb{E}}^{wc}_\theta(N) \lesssim \frac{s \log^2(ep)}{\vartheta^2}$$

and

$$\bar{\mathbb{E}}_\theta(N) \lesssim \frac{(s \wedge \sqrt{p}) \log^2(ep)}{\vartheta^2}.$$
Comparison with other methods

We compare ocd with other recently proposed methods:

- Mei: $\ell_1$ and $\ell_\infty$ aggregation of likelihood ratio tests in each coordinate. (Mei, 2010)
- XS: Use window-based method to aggregate statistics for testing the null against a normal mixture in each coordinate. (Xie and Siegmund, 2013)
- Chan: Similar to XS, but with an improved choice of tuning parameters. (Chan, 2017)

Simulation settings: $p \in \{100, 2000\}$, $s \in \{5, \lceil \sqrt{p} \rceil, p\}$, $\vartheta \in \{1, 0.5, 0.25\}$ and $\theta$ is generated as $\vartheta U$, where $U$ is uniformly distributed on the union of all $s$ sparse unit spheres in $\mathbb{R}^p$.

- All thresholds are determined using Monte Carlo simulation.
Comparison with other methods

<table>
<thead>
<tr>
<th>$p$</th>
<th>$s$</th>
<th>$\vartheta$</th>
<th>ocd</th>
<th>Mei</th>
<th>XS</th>
<th>Chan</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>5</td>
<td>1</td>
<td>46.9</td>
<td>125.9</td>
<td>47.3</td>
<td>42.0</td>
</tr>
<tr>
<td>100</td>
<td>5</td>
<td>0.5</td>
<td>174.8</td>
<td>383.1</td>
<td>194.3</td>
<td>163.7</td>
</tr>
<tr>
<td>100</td>
<td>5</td>
<td>0.25</td>
<td>583.5</td>
<td>970.4</td>
<td>2147</td>
<td>1888.8</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>1</td>
<td>53.8</td>
<td>150.1</td>
<td>52.9</td>
<td>51.5</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>0.5</td>
<td>194.4</td>
<td>458.2</td>
<td>255.8</td>
<td>245.6</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>0.25</td>
<td>629.7</td>
<td>1171.3</td>
<td>2730.7</td>
<td>2484.9</td>
</tr>
<tr>
<td>100</td>
<td>100</td>
<td>1</td>
<td>74.4</td>
<td>268.3</td>
<td>89.6</td>
<td>102.1</td>
</tr>
<tr>
<td>100</td>
<td>100</td>
<td>0.5</td>
<td>287.9</td>
<td>834.9</td>
<td>526.8</td>
<td>756.0</td>
</tr>
<tr>
<td>100</td>
<td>100</td>
<td>0.25</td>
<td>1005.8</td>
<td>1912.9</td>
<td>3598.3</td>
<td>3406.6</td>
</tr>
<tr>
<td>2000</td>
<td>5</td>
<td>1</td>
<td>67.3</td>
<td>316.7</td>
<td>79.5</td>
<td>59.5</td>
</tr>
<tr>
<td>2000</td>
<td>5</td>
<td>0.5</td>
<td>247.3</td>
<td>680.2</td>
<td>607.7</td>
<td>285.0</td>
</tr>
<tr>
<td>2000</td>
<td>5</td>
<td>0.25</td>
<td>851.3</td>
<td>1384.8</td>
<td>4459.2</td>
<td>3856.9</td>
</tr>
<tr>
<td>2000</td>
<td>44</td>
<td>1</td>
<td>136.0</td>
<td>596.1</td>
<td>149.1</td>
<td>145.0</td>
</tr>
<tr>
<td>2000</td>
<td>44</td>
<td>0.5</td>
<td>479.1</td>
<td>1270.8</td>
<td>2945.5</td>
<td>2751.4</td>
</tr>
<tr>
<td>2000</td>
<td>44</td>
<td>0.25</td>
<td>1584.2</td>
<td>2428.8</td>
<td>4457.8</td>
<td>5049.7</td>
</tr>
<tr>
<td>2000</td>
<td>2000</td>
<td>1</td>
<td>360.7</td>
<td>2126.5</td>
<td>1020.0</td>
<td>2074.7</td>
</tr>
<tr>
<td>2000</td>
<td>2000</td>
<td>0.5</td>
<td>1296.0</td>
<td>3428.1</td>
<td>4669.3</td>
<td>4672.7</td>
</tr>
<tr>
<td>2000</td>
<td>2000</td>
<td>0.25</td>
<td>3436.7</td>
<td>4140.4</td>
<td>5063.7</td>
<td>5233.5</td>
</tr>
</tbody>
</table>

**Table:** Estimated response delay for ocd, Mei, XS and Chan over 200 repetitions, with $z = 0$ and $\gamma = 5000$. 
We propose a new, multiscale method for high-dimensional online changepoint detection.

We perform likelihood ratio tests against simple alternatives of different scales in each coordinate, and aggregate these statistics.

R package `ocd` is available on CRAN.

Main reference

References

References

Thank you!