Online Centralized Non-parametric Change-point Detection via Graph-based Likelihood-ratio Estimation

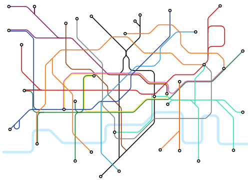
Alejandro de la Concha   Nicolas Vayatis
Argyris Kalogeratos

Université Paris-Saclay, ENS Paris-Saclay, CNRS, Centre Borelli, France

Contents

Motivation

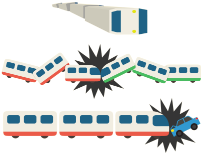
Complex systems generating real-time data are everywhere.



[1] Source: Shutterstock

Motivation

A change-point detection may have different interpretations.

[1]Source: Shutterstock

Previous work

Classical change-point detection frameworks [Aminikhanghahi and Cook, 2016]

Classification according to the way data becomes availble:

- *Offline detection*: [Truong et al., 2020]
- *Online detection*: [Tartakovsky et al., 2014, Xie et al., 2021]

Detection in sensor networks

Goal: How to efficiently send information from local sensors to a fusion center aiming to spot a global change-point?

1. Online decentralized and synchronous detection:
   [Veeravalli, 2001, Tartakovsky and Veeravalli, 2002]
   [Mei, 2005, Tartakovsky and Veeravalli, 2008]
2. Online decentralized with delay between sensors:
   [Hadjiliadis et al., 2009, MEI, 2010, Raghavan and Veeravalli, 2010]

All works based on parametric techniques!

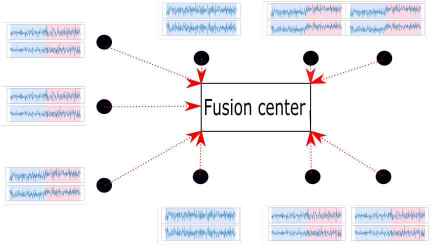
Network of sensors

Setting: N data streams, let $x_{v,t}$ be the observation at stream v at time t.
Problem: At a timestamp τ, the distribution associated with the streams observed at nodes belonging to a set C, changes:

$$\begin{cases} t < \tau & x_{v,t} \sim p_v; \\ t \geq \tau + \Delta_v & x_{v,t} \sim q_v; \end{cases}$$

where $p_v \neq q_v$ if $v \in C$, otherwise $p_v = q_v$. p_v and q_v are in most cases known parametric pdfs.

Goal: How to efficiently send local information to a fusion center and detect τ?

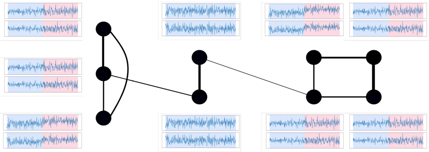


Recent work

Setting: N synchronous data streams, each generated by a node of a graph G. $x_{v,t} \in R^d$ is the observation at node v at time t.

Problem: How to integrate the a priori information provided by G into the detection task?

Goal: Identify τ and C.



Recent work

[Zou and Veeravalli, 2018, Zou et al., 2019]

- Online centralized detection with delays.
- p_θ, q_θ, $\theta \in F_\Theta$, where F_Θ is a known parametric family.
- The change-point will propagate along the edges of G.

[de la Concha Duarte et al., 2021]

- Offline centralized detection with synchronicity.
- Change in the mean of a multivariate Gaussian distribution.
- The covariance between observations is explained mainly by G.

Recent work

[Ferrari and Richard, 2020]

- Online decentralized detection with synchronicity
- Non-parametric approach based on likelihood-ratio estimation
- G has clusters and one of them will be affected

Online Centralized Kernel- and Graph-based change-point detection (OCKGD) [de la Concha et al., 2022, 2023]

- Online centralized change-point detection with synchronicity
- Non-parametric based on likelihood-ratio estimation
- Smoothness of likelihood-ratio estimates

Contents

Likelihood-ratio

Definition

Consider a feature space $\mathcal{X} \subset \mathbb{R}^n$ and two probability distributions P and Q ($Q \ll P$) such that they admit density functions $p(x)$ and $q(x)$ with respect to dx, then the **likelihood-ratio** $r(x) = \frac{q(x)}{p(x)}$ $x \in \mathcal{X}$.

Relative likelihood-ratio

For $\alpha \in (0, 1]$

$$r^\alpha(x) = \frac{q(x)}{(1-\alpha)p(x) + \alpha q(x)} = \frac{q(x)}{p^\alpha(x)} \leq \frac{1}{\alpha}.$$

Basic tool in change-point detection: Schewart Chart, CUSUM, Shriryaev-Robets Procedure, GLR [Tartakovsky et al., 2014].

f-divergence

Definition

A f-divergence (ϕ-divergence) is a similarity measure between two probability distributions that are described by the p and q, over the input space $\mathcal{X} \subset \mathbb{R}^d$:

$$\mathcal{D}_\phi(p,q) = \int p(x)\,\phi\big(q(x)/p(x)\big)\,dx, \qquad \text{for } x \in \mathcal{X},$$

where $f : \mathbb{R} \to \mathbb{R}$ is a convex and semi-continuous real function such that $f(1) = 0$ [Csiszár, 1967].

- $\phi(x) = -\log(x)$ Kullback–Leibler's Divergence [Kullback, 1959]. (Optimality results in online change-point detection)
- $\phi(x) = \frac{(x-1)^2}{2}$, we identify the Pearson's Divergence [Pearson, 1900].

Reproducing Kernel Hilbert Space

Definition

Reproducing Kernel Hilbert Space (RKHS) $\mathbb{H}$ containing functions $f : \mathcal{X} \to \mathbb{R}$. The space is equipped with the inner product $\langle \cdot, \cdot \rangle_{\mathbb{H}} : \mathbb{H} \times \mathbb{H} \to \mathbb{R}$, which is reproduced by a symmetric and positive semi-definite kernel function $K(\cdot, \cdot) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, that satisfies:

- $\langle K(\cdot, x), f \rangle_{\mathbb{H}} = f(x)$, for any $f \in \mathbb{H}$;
- $\mathbb{H} = \overline{\mathrm{span}(\{K(x, \cdot) : \forall x \in \mathcal{X}\})}$, where span refers to all the linear combinations of $K(x, \cdot)$.

Graph-related concepts

- $G = (V, E, W)$ a positive weighted and undirected graph:
  V set of vertices, E set of edges, $W \in \mathbb{R}^{N\times N}$ adjacency matrix.
- $W_{u,v}$ reflects the strenght of the relationship between nodes $u, v \in V$.
- Combinatorial Laplacian of G: $\mathcal{L} = \mathrm{diag}((d_v)_{v\in V}) - W$.

Definition

Given a set of vectors associated with each node $v \in V$ $\theta_1, ..., \theta_N$, $\theta_v \in \mathbb{R}^L$ and the concatenation of parameters $\Theta = (\theta_1, ..., \theta_N)$. The smoothnes of Θ is defined by:

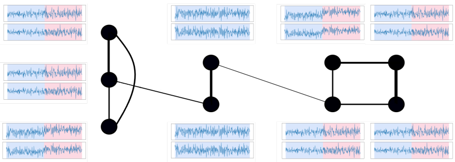
$$\mathcal{S}(\Theta) = \Theta^\top (\mathcal{L} \otimes \mathbb{I}_L)\, \Theta = \sum_{u,v\in V} W_{u,v} \left\|\theta_u - \theta_v\right\|^2 . \quad (1)$$

Contents

Our setting

Working hypotheses

- N synchronous data streams, each generated by a node of a connected graph G.
- $x_{v,t} \in R^d$ be the observation at node v at time t. All from the same input space $x_{v,t} \in \mathcal{X}$ $\forall v \in V, t \in \mathcal{N}$. Observations are independent in time.
- The vector of the relative likelihood-ratios $r^{\alpha}(X) = (r_1^{\alpha}(x), ..., r_N^{\alpha}(x)) \in \mathbb{R}^N$ is a smooth signal over G.



General idea

OCKG

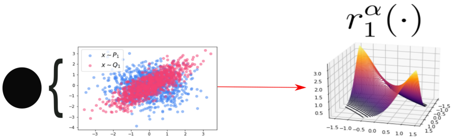
The proposed algorithm is mainly based on three components:

1. **Estimation:** Non-parametric estimation of the vector of relative likelihood-ratios $r^{\alpha}{}_t(\cdot) = (r^{\alpha}{}_{1,t}(\cdot), ..., r^{\alpha}{}_{N,t}(\cdot))$ as observations arrive.
2. **Detection:** Use the estimators to produce node-level scores $\{S_{v,t}\}_{v \in V}$ and a global-level score S_t for all the system.
3. **Identification:** Use the node-level scores $\{S_{v,t}\}_{v \in V}$ to identify C.

Machine Learning and change-point detection

Likelihood-ratio estimation

- Given two sets of observations $x_1, ..., x_n \sim p(\cdot)$ and $x'_1, ..., x'_n \sim q(\cdot)$.
- $r(\cdot)$ is the solution of an optimization problem defined in terms of a functional space $\mathbb{H}$ (RKHS) [Nguyen et al., 2008, Sugiyama et al., 2012]



Theoretical justification

Lemma

For any class of functions $\mathcal{F}$ mapping from $\mathcal{X}$ to $\mathbb{R}$, we have the lower bound for the ϕ-divergence:

$$D_\phi(p,q) \geq \sup_{f \in \mathcal{F}} \int \big[f(x)q(x)dx - \phi^*(f(x))p(x) \big].$$

Equality holds if and only if the subdifferential $\partial\phi\left(\frac{q(x)}{p(x)}\right)$ contains an element of $\mathcal{F}$, where ϕ^* is the conjugate dual function of ϕ, i.e. $\phi^*(v) = \sup_{u \in \mathbb{R}}[uv - \phi(u)]$ [Nguyen et al., 2008]

Pearson-Divergence $\phi(x) = \frac{(x-1)^2}{2}$ and $F = \mathbb{H}$

$\min_{f \in \mathbb{H}} \int \frac{f^2(x)}{2} p(x)dx - \int f(x)q(x)dx = \int \frac{(f(x)-r(x))^2}{2} p(x)dx$

Likelihood-ratio estimation on graphs

Framework

- Infer the node-level relative likelihood-ratios $r_v^\alpha(\cdot) = \frac{q_v(\cdot)}{(1-\alpha)p_v(\cdot)+\alpha q_v(\cdot)}$
- Non-parametric estimation based on a kernel function $K(\cdot,\cdot) : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^+$ to avoid making hypotheses on the nature of $q_v(\cdot)$ and $p_v(\cdot)$
- Capitalize over the information provided apriori by the graph: $\|r_u - r_v\|_{\mathbb{H}} < \epsilon$ if $W_{uv} \neq 0$

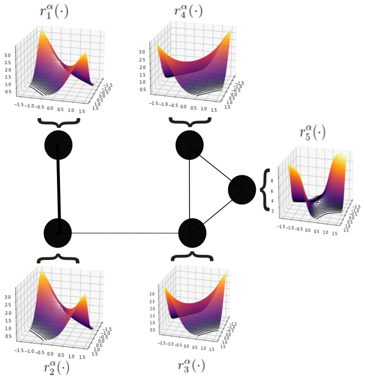


Figure: We aim to estimate the set of functions $\{r_v^\alpha(\cdot)\}_{v\in V}$ by exploiting $G = (V, E, W)$.

Multitasking formulation

By the reproduction property of $\mathbb{H}$, for each $v \in V$, f_v takes the form:

$$f_v(x) = \sum_{l=1}^{L} \theta_{v,i} K(x, x_i)$$

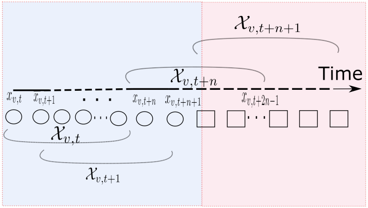
Cost-function

$$\min_{\Theta \in \mathbb{R}^{NL}} \overbrace{\frac{1}{N} \sum_{v \in V} \frac{\mathbb{E}_{p_v^\alpha(x)}[(r_v^\alpha(x) - f_v(x))^2]}{2}}^{\text{Pearson-Divergence based node-level}} + \overbrace{\frac{\lambda\gamma}{2} \sum_{v \in V} \|\theta_v\|^2}^{\text{node-level regularization term}}$$
$$+ \overbrace{\frac{\lambda}{2} \sum_{u,v \in V} W_{uv} \|\theta_u - \theta_v\|^2}^{\text{graph-level regularization term}}.$$

Optimization problem

Sliding windows: At each time t we compare two time-adjacent samples of size n:

$$\mathcal{X}_{v,t} = [x_{v,t-n},\, x_{v,t-(n-1)},\, ...,\, x_{v,t}]$$
$$\mathcal{X}_{v,t+n} = [x_{v,t+1},\, x_{v,t+2},\, ...,\, x_{v,t+n}]$$



Useful quantities

$$\phi(x) = (K(x, x_1), ..., K(x, x_L))$$
$$H_{v,t} = \frac{1}{n} \sum_{x \in \mathcal{X}_{v,t}} \phi(x)\phi(x)^\top, \quad H'_{v,t} = \frac{1}{n} \sum_{x \in \mathcal{X}_{v,t+n}} \phi(x)\phi(x)^\top, \quad h'_{v,t} = \frac{1}{n} \sum_{x \in \mathcal{X}_{v,t+n}} \phi(x)$$

Practical implementation

Penalized cost function

$$\min_{\Theta \in \mathbb{R}^{NL}} \frac{1}{N} \sum_{v \in V} \overbrace{\left((1-\alpha) \frac{\theta_v^\top H_{v,t} \theta_v}{2} + \alpha \frac{\theta_v^\top H'_{v,t} \theta_v}{2} - h'_{v,t} \theta_v \right)}^{\text{Pearson-divergence node-level}} + \overbrace{\frac{\lambda \gamma}{2} \sum_{v \in V} \|\theta_v\|^2}^{\text{node-level regularization term}}$$

$$+ \overbrace{\frac{\lambda}{4} \sum_{u,v \in V} W_{uv} \|\theta_u - \theta_v\|^2}^{\text{graph-level regularization term}} = \min_{\Theta \in \mathbb{R}^{NL}} \frac{\Theta^\top (A_t) \Theta}{2} - \Theta^\top b_t$$

Question: How to estimate the parameters Θ_t online?

- Optimization procedure that can take advantage of the previous estimate Θ_{t-1} and scales nicely with the number of nodes.
- Dictionary building strategy to integrate new observations.
- Hyperparameters selection strategy: Kernel parameter σ^2, penalization constants $\lambda, \gamma > 0$.

Cyclic Block Coordinate Gradient Descent (CBCGD)

Theorem

[Li et al., 2018] Suppose that for a dictionary D of size $L \geq 2$ and the update with respect to the node parameter θ_v at the i-th cycle is computed by:

$$\hat{\theta}^{(i)}_{v,t} = \overbrace{\frac{\eta_{v,t}}{\eta_{v,t}+\lambda\gamma}\hat{\theta}^{(i-1)}_{v,t} - \frac{1}{\eta_{v,t}+\lambda\gamma}\left[\left(\frac{(1-\alpha)H_{v,t}+\alpha H'_{v,t}}{N}\right)\hat{\theta}^{(i-1)}_{v,t} - \frac{h'_{v,t}}{N}\right]}^{\text{component depending on node } v}$$

$$\overbrace{-\frac{\lambda}{\eta_{v,t}+\lambda\gamma}\left[d_v\hat{\theta}^{(i-1)}_{v,t} - \sum_{u\in ng(v)} W_{uv}\left(\hat{\theta}^{(i)}_{u,t}\mathbf{1}_{u<v} + \hat{\theta}^{(i-1)}_{u,t}\mathbf{1}_{u\geq v}\right)\right]}^{\text{component depending on the graph}}$$

Then, if we fix the learning rate for node v at $\eta_{v,t} = e_{\max}\left(\frac{(1-\alpha)}{N}H_{v,t} + \frac{\alpha}{N}H'_{v,t} + \lambda d_v \mathbb{I}_L\right)$, then the required number of interactions to achieve accuracy level $\epsilon > 0$:

$$i_{\max} = \left[\frac{\lambda\gamma(M_{\min}+\lambda\gamma) + 16M^2\log^2(3NL)}{\lambda\gamma(M_{\min}+\lambda\gamma)}\log\left(\frac{\Phi(\Theta^{(0)}_t)-\Phi(\Theta^*_t)}{\epsilon}\right)\right]$$

Dictionary building strategy [Richard et al., 2009]

Question: How to integrate new elements into the dictionary?

Coherence: Measures the linear dependencies between elements of a dictionary:

$$\mu = \max_{l \neq l'} \left| \langle K(x_l, \cdot), K(x_{l'}, .) \rangle_{\mathbb{H}} \right| = \max_{l \neq l'} \left| K(x_l, x_{l'}) \right| .$$

Algorithm: Dictionary update

```
Incomming observation x_{v,t+n-1}
if max_{l∈{1,...,L}} K(x_{v,l}, x_{v,t+n-1}) ≤ μ_0 then
    Add x_{v,t+n-1} to the dictionary D
    if the maximum dictionary size is reached, delete the datapoint with
      the highest coherence
```

Hyperparameter selection

Question: How to chose the Kernel parameters σ and the penalization terms $\lambda, \gamma > 0$?
Cross-validation based on the score:

$$\hat{\ell}(\sigma, \lambda, \gamma) = \frac{1}{R} \sum_{r=1}^{R} \ell^{(r)}(\Theta)$$

where R is the number of splits:

$$\ell^{(r)}(\hat{\Theta}(\gamma, \lambda)) = \frac{1}{N} \sum_{v \in V} (1 - \alpha) \frac{\hat{\theta}_v^\top H_{v,\text{test}} \hat{\theta}_v}{2} + \alpha \frac{\hat{\theta}_v^\top H'_{v,\text{test}} \hat{\theta}_v}{2} - h'_{v,\text{test}} \theta_v$$

$\sigma^*, \lambda^*, \gamma^* = \text{argmin}_{\sigma,\lambda,\gamma} \hat{\ell}(\sigma, \lambda, \gamma)$

Pearson-Divergence estimation

$$
\begin{aligned}
PE_v(p_v^\alpha, q_v) &= r_v^\alpha(x) - \frac{(1-\alpha)}{2} r_v^\alpha(x)^2 - \frac{\alpha}{2} r_v^\alpha(x)^2 - \frac{1}{2} \\
&\approx \hat{PE}_v^\alpha(\mathbf{X_v}, \mathbf{X'_v}) = -\left((1-\alpha)\frac{\hat{\theta}_v^\top H_v \hat{\theta}_v}{2} + \alpha \frac{\hat{\theta}_v^\top H'_v \hat{\theta}_v}{2} - h'_v \hat{\theta}_v\right) - \frac{1}{2} \\
&= -\ell_v(\hat{\theta}_v) - \frac{1}{2}.
\end{aligned} \tag{2}
$$

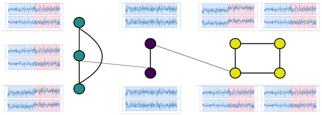
1. $PE(p,q) = 0$ if and only if $p = q$
2. Θ_t minimizes $\frac{1}{N}\left(\sum_{v \in V} \ell_{v,t}(\theta_{v,t}) \right) + \frac{\lambda}{2} \Theta_t^\top \left([\mathcal{L} + \gamma \mathbb{I}_N] \otimes \mathbb{I}_L \right) \Theta_t$
3. PE is NOT symmetric!!!

Detection

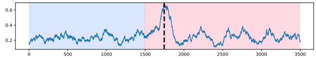
Question: When do we raise a global alarm?

- Estimate $\vec{\theta}_{v,t}$ using $\mathcal{X}_{v,t}$ and $\mathcal{X}_{v,t+n}$ and viceversa $\overleftarrow{\theta}_{v,t}$ with $\mathcal{X}_{v,t+n}$ and $\mathcal{X}_{v,t}$.
- For each node estimate the quantities $\hat{PE}_v^{\alpha}(\mathcal{X}_{v,t}, \mathcal{X}_{v,t+n})$ and $\hat{PE}_v^{\alpha}(\mathcal{X}_{v,t+n}, \mathcal{X}_{v,t})$
- Node-level score: $S_{v,t} = \left(\hat{PE}_v^{\alpha}(\mathcal{X}_{v,t}, \mathcal{X}_{v,t+n}) + \hat{PE}_v^{\alpha}(\mathcal{X}_{v,t+n}, \mathcal{X}_{v,t})\right)^+$
- Global-level score: $S_t = \sum_{v \in V} S_{v,t}$. If $S_t > \eta$ raise an alarm.

Node-level scores



Global-level score



Identification

Question: How to identify the nodes in C ? Select the nodes such that $S_{v,t} \geq \eta_v$.

Question: How to find the parameter η_v?
Potential solution: **Permutation test** de la Concha et al. [2022]
Select η_v such that $\mathbb{P}(S_v \geq \eta_v | \text{No change-point}) \leq \alpha$.

Summary

Algorithm: The OCKG detector

$\vec{\Theta}_n^{(0)} = \overleftarrow{\Theta}_n^{(0)} = LN$

■ **Online estimation and detection**

for $t \in \{n, \ldots, \}$ **do**

  for $v \in \{1, \ldots, N\}$ **do**

    Observe $x_{v,t+n-1}$ and update the sliding windows $\mathcal{X}_t, \mathcal{X}_{t+n}$ (Eq. 22)

    ☐ **Dictionary update**

    if $\max_{l\in\{1,\ldots,L_1\}} k(x_{v,l}, x_{v,t+n-1}) \leq \mu_0$ **then**

      Add $x_{v,t+n-1}$ to the dictionary D_1 if the maximum dictionary size is reached, delete the datapoint with the highest coherence

  ☐ **Parameters update**

  Define $\vartheta_v = [\theta_{v,t-1}^\top, d_1]$, ($d_1$ is the number of new elements added to the dictionary) Initialize $\vec{\theta}_{v,t-1}^{(0)} = \vartheta_v$

  Fix $\mathcal{X} = \mathcal{X}_t$ and $\mathcal{X}' = \mathcal{X}_{t+n}$

  for $v \in \{1, \ldots, N\}$ **do**

    Compute the quantities H_v, H'_v, h'_v.

    Fix $\gamma_{v,t} = e_{\max}\left(\frac{(1-\alpha)H_{v,t}+\alpha H'_{v,t}}{N} + \lambda d_v \mathbb{I}_{L_1}\right)$

  while $\left\|\vec{\Theta}_t^{(i)} - \vec{\Theta}_t^{(i-1)}\right\| > \epsilon$ **do**

    for $v \in \{1, \ldots, N\}$ **do**

      Update $\vec{\theta}_v^{(i)}$ via CBCGD

for $t \in \{n, \ldots, \}$ **do**

  for $v \in \{1, \ldots, N\}$ **do**

    Estimate $\hat{PE}_v^{\alpha}(\mathcal{X}_v, \mathcal{X}'_v)$

  Fix $\mathcal{X} = \mathcal{X}_{t+n}$ and $\mathcal{X}' = \mathcal{X}_t$

  Repeat previous steps to compute $\overleftarrow{\Theta}_t$ and $\hat{PE}_v^{\alpha}(\mathcal{X}'_{v,t}, \mathcal{X}_{v,t})$

  ☐ **Online detection and Identification**

  Compute the node scores $S_{v,t}$ Compute the global score $S_t = \sum_{v \in V} S_{v,t}$

  if $S_t > \eta$ **then**

    A change-point is detected at $\hat{\tau} = t$

    if $S_{v,t} > \eta_v$ **then**

      Add v to $\hat{C}$

  Return $\hat{\tau}$ and $\hat{C}$

Contents

Algorithms

Nougat [Ferrari and Richard, 2020]:

1. Update the quantity $g_{v,t}(\cdot) = \hat{r}_{v,t}(\cdot) - 1$ ($\hat{r}_{v,t}$ unreguralized likelihood-ratio via a SGD step.
2. Compute log-likelihood-ratio independently for each node: $l_{v,t}(x_{v,t}) = log(g_{v,t-1}(x_{v,t}) + 1)$.
3. Integrate the graph information via a graph filter $\hat{l}_t = \mathbf{L}(l_t)$ (L: Graph Fourier Scan Statistic [Sharpnack et al., 2016]).
4. Node level score: $S_{v,t} = \left|\hat{l}_v\right|$, Global level score: $S_t = \left\|\hat{l}_t\right\|^2$.

Pool:

OCKGD with $W = 0$ (no-connections), equivalent to RULSIF [Liu et al., 2013].

Experiment I.A

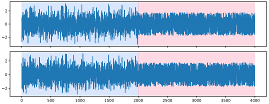
Case I. Changes in node clusters
Stochastic Block Model with 4 clusters, $C1$,...,$C4$, each containing 20 nodes.

A. Bivariate Gaussian distribution to Gaussian copula with uniform marginals

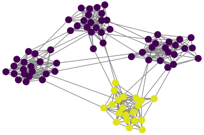
$$(x, y) \sim N(\mu, \Sigma), \mu = (0, 0), \Sigma_{x,x} = 1, \Sigma_{x,y} = \frac{4}{5}$$

$$\rightarrow (x, y) \sim GC, \quad \Sigma_{x,x} = 1, \Sigma_{x,y} = \frac{4}{5}.$$

Time-series with change-point



Graph structure



Results

Scenario	Detector	Detection delay (std)	AUC (std)	Precision
Experiment I.A n =125	OCKG $\alpha = 0.1$	126.26 (11.95)	**0.89 (0.05)**	**1.00**
	OCKG $\alpha = 0.5$	129.67 (11.37)	0.85 (0.06)	0.98
	Pool $\alpha = 0.1$	123.72 (24.08)	0.82 (0.05)	0.58
	Pool $\alpha = 0.5$	131.90 (21.02)	0.80 (0.05)	0.80
	Nougat	146.50 (70.74)	0.56 (0.22)	0.12
Experiment I.A n =250	OCKG $\alpha = 0.1$	252.72 (14.82)	**0.93 (0.03)**	**1.00**
	OCKG $\alpha = 0.5$	251.31 (21.82)	0.89 (0.04)	0.98
	Pool $\alpha = 0.1$	249.54 (25.20)	0.86 (0.04)	0.92
	Pool $\alpha = 0.5$	245.32 (22.20)	0.87 (0.04)	0.94
	Nougat	273.50 (96.59)	0.67 (0.19)	0.20
Experiment I.A $n = 500$	OCKG $\alpha = 0.1$	502.70 (6.49)	**0.99 (0.00)**	**1.00**
	OCKG $\alpha = 0.5$	500.84 (5.15)	**0.99 (0.00)**	**1.00**
	Pool $\alpha = 0.1$	506.20 (18.31)	**0.99 (0.00)**	**1.00**
	Pool $\alpha = 0.5$	501.90 (7.86)	**0.99 (0.00)**	**1.00**
	Nougat	576.86 (129.27)	0.66 (0.20)	0.74

Experiment I.B

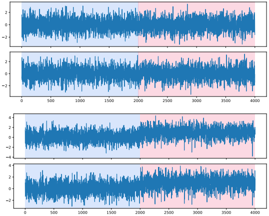
I. Changes in node clusters

Stochastic Block Model with 4 clusters, $C1$,...,$C4$, each containing 20 nodes.

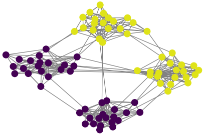
B. Change in the covariance matrix or mean vector of bivariate Gaussian distribution

$$\begin{cases} \mu = (0,0), \Sigma_{1,2} = \frac{4}{5} & \rightarrow \quad \mu = (0,0), \Sigma_{1,2} = \frac{-4}{5} \quad \text{if } C1 \\ \mu = (0,0), \Sigma_{1,2} = \frac{4}{5} & \rightarrow \quad \mu = (0,0), \Sigma_{1,2} = 0 \quad \text{if } C2 \\ \mu = (0,0), \Sigma_{1,2} = \frac{-4}{5} & \rightarrow \quad \mu = (0,0), \Sigma_{1,2} = 0 \quad \text{if } C3 \\ \mu = (0,0), \Sigma_{1,2} = \frac{4}{5} & \rightarrow \quad \mu = (1,1), \Sigma_{1,2} = \frac{4}{5} \quad \text{if } C4. \end{cases}$$

Time-series with change-point



Graph structure



Results

Scenario	Detector	Detection delay	(std)	AUC (std)	Precision
Experiment I.B n =25	OCKG $\alpha = 0.1$	25.04	(0.20)	**0.99 (0.02)**	**1.00**
	OCKG $\alpha = 0.5$	25.06	(0.31)	0.98 (0.02)	**1.00**
	Pool $\alpha = 0.1$	25.12	(0.48)	0.91 (0.05)	**1.00**
	Pool $\alpha = 0.5$	25.02	(0.24)	0.96 (0.03)	**1.00**
	Nougat	40.68	(4.12)	0.76 (0.17)	1.00
Experiment I.B $n = 50$	OCKG $\alpha = 0.1$	50.18	(0.55)	**1.00 (0.00)**	**1.00**
	OCKG $\alpha = 0.5$	50.16	(0.54)	0.99 (0.01)	**1.00**
	Pool $\alpha = 0.1$	50.02	(0.32)	0.93 (0.06)	**1.00**
	Pool $\alpha = 0.5$	50.00	(0.20)	**1.00 (0.00)**	**1.00**
	Nougat	67.98	(7.63)	0.78 (0.17)	0.92
Experiment I.B n =100	OCKG $\alpha = 0.1$	100.12	(0.47)	**1.00 (0.00)**	**1.00**
	OCKG $\alpha = 0.5$	100.02	(0.14)	**1.00 (0.00)**	**1.00**
	Pool $\alpha = 0.1$	100.06	(0.24)	**1.00 (0.01)**	**1.00**
	Pool $\alpha = 0.5$	100.00	(0.00)	**1.00 (0.00)**	**1.00**
	Nougat	124.23	(13.50)	0.88 (0.09)	1.00

Experiment II.A

Case II. Changes in set of connected nodes

Barabási-Albert model with 100 nodes. C by selecting a node at random an k-hop.

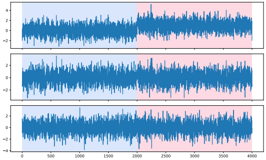
A.Shift in the mean on one of the cluster components

$$x_v \sim N(\mu, \Sigma), \mu = (0,0,0), \Sigma_{i,i} = 1, \Sigma_{1,2} = \frac{4}{5}, \Sigma_{3,1} = 0$$

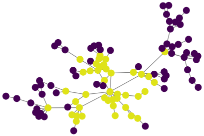
$$\rightarrow x_v \sim N(\mu, \Sigma), \mu = (1,0,0)$$

$$\Sigma_{i,i} = 1, \Sigma_{1,2} = \frac{4}{5}, \Sigma_{3,1} = 0.$$

Time-series with change-point



Graph structure



Results

Scenario	Detector	Detection delay (std)		AUC (std)	Precision
Experiment II.A n =25	OCKG $\alpha = 0.1$	25.44	(1.96)	**0.97 (0.02)**	**1.00**
	OCKG $\alpha = 0.5$	25.06	(1.34)	**0.97 (0.02)**	0.96
	Pool $\alpha = 0.1$	24.51	(1.68)	0.91 (0.03)	0.82
	Pool $\alpha = 0.5$	24.44	(2.03)	0.93 (0.02)	0.86
	Nougat	34.25	(13.80)	0.64 (0.17)	0.08
Experiment II.A n =50	OCKG $\alpha = 0.1$	50.38	(1.21)	**0.99 (0.01)**	**1.00**
	OCKG $\alpha = 0.5$	50.67	(1.48)	0.96 (0.04)	0.98
	Pool $\alpha = 0.1$	48.55	(5.14)	0.91 (0.04)	0.98
	Pool $\alpha = 0.5$	49.63	(2.38)	**0.99 (0.01)**	0.98
	Nougat	77.84	(10.24)	0.75 (0.17)	0.76
Experiment II.A n =100	OCKG $\alpha = 0.1$	100.52	(1.25)	0.99 (0.00)	**1.00**
	OCKG $\alpha = 0.5$	100.16	(0.64)	**1.00 (0.00)**	**1.00**
	Pool $\alpha = 0.1$	99.86	(1.23)	0.99 (0.00)	**1.00**
	Pool $\alpha = 0.5$	100.38	(1.01)	0.99 (0.00)	**1.00**
	Nougat	127.52	(13.87)	0.77 (0.16)	0.88

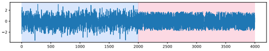
Experiment II.B

Case II. Changes in set of connected nodes
Barabási-Albert model with 100 nodes. C by selecting a node at random an k-hop.

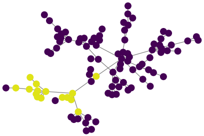
B. Change in probability law

$$x_v \sim N(0,1) \;\rightarrow\; x_v \sim U(-\sqrt{3},\sqrt{3}) \text{ if } v \in C.$$

Time-series with change-point



Graph structure



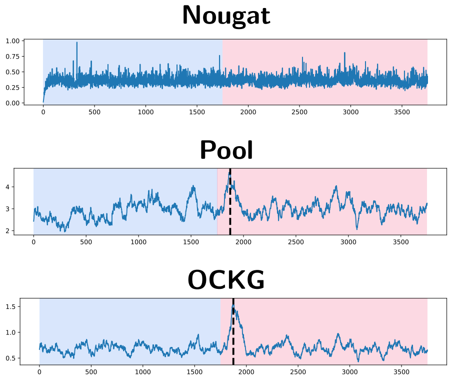
Results

Scenario	Detector	Detection delay (std)	AUC (std)	Precision
Experiment II.B n =125	OCKG $\alpha = 0.1$	128.17 (9.91)	**0.86 (0.04)**	0.94
	OCKG $\alpha = 0.5$	129.63 (16.66)	**0.86 (0.04)**	**0.96**
	Pool $\alpha = 0.1$	130.05 (12.48)	0.79 (0.06)	0.88
	Pool $\alpha = 0.5$	133.64 (15.57)	0.85 (0.05)	**0.96**
	Nougat	146.08 (17.78)	0.54 (0.19)	0.26
Experiment II.B n =250	OCKG $\alpha = 0.1$	249.96 (20.19)	**0.96 (0.02)**	**1.00**
	OCKG $\alpha = 0.5$	251.12 (15.29)	0.88 (0.04)	**1.00**
	Pool $\alpha = 0.1$	258.51 (12.18)	0.92 (0.03)	0.98
	Pool $\alpha = 0.5$	254.58 (19.09)	0.88 (0.04)	1.00
	Nougat	331.00 (121.15)	0.49 (0.17)	0.34
Experiment II.B n =500	OCKG $\alpha = 0.1$	498.70 (8.88)	**1.00 (0.00)**	**1.00**
	OCKG $\alpha = 0.5$	500.00 (1.80)	**1.00 (0.00)**	**1.00**
	Pool $\alpha = 0.1$	497.94 (15.47)	**1.00 (0.00)**	**1.00**
	Pool $\alpha = 0.5$	500.12 (2.28)	**1.00 (0.00)**	**1.00**
	Nougat	529.86 (68.53)	0.39 (0.16)	0.98

Contents

Conclusions

- OCKG and Pool have the best performance. Nougat induces noise by the stochastic-gradient descent strategy.
- The window size n induces a bias for all the methods. Detection time $\hat{\tau} = \tau + n$.
- OCKG performs better than Pool when the window size is smaller. Lower detection delay.
- OCKG generates cleaner node level-scores.



Open questions

- Lack of real-examples in the literature.
- Lack of theoretical guarantees to relate the choice of η and false alarms in LRE-based methods.
- The hypothesis of synchronicity of node-level change-points is not satisfied for many applications.

Preprints

Further details:

- de la Concha, A., Kalogeratos, A., Vayatis, N. (2022). Collaborative likelihood-ratio estimation over graphs (Version 1). arXiv. ▸ Link
- de la Concha, A., Kalogeratos, A., Vayatis, N. (2023). Online Centralized Non-parametric Change-point Detection via Graph-based Likelihood-ratio Estimation. arXiv. ▸ Link

Thank you

References I

S. Aminikhanghahi and D. Cook. A survey of methods for time series change point detection. *Knowledge and Information Systems*, 51: 339–367, 2016.

Imre Csiszár. On topological properties of f-divergences. *Studia Scientiarum Mathematicarum Hungarica*, 2:329–339, 1967.

Alejandro de la Concha, Argyris Kalogeratos, and Nicolas Vayatis. Collaborative likelihood-ratio estimation over graphs, 2022.

Alejandro de la Concha, Argyris Kalogeratos, and Nicolas Vayatis. Online centralized non-parametric change-point detection via graph-based likelihood-ratio estimation, 2023. URL https://arxiv.org/abs/2301.03011.

Alejandro de la Concha Duarte, Nicolas Vayatis, and Argyris Kalogeratos. Offline detection of change-points in the mean for stationary graph signals. In *Proceedings of International Conference on Artificial Intelligence and Statistics*, volume 130, pages 3430–3438, 2021.

A. Ferrari and C. Richard. Non-parametric community change-points detection in streaming graph signals. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5545–5549, 2020.

Olympia Hadjiliadis, Hongzhong Zhang, and H. Vincent Poor. One shot schemes for decentralized quickest change detection. *IEEE Trans. Inf. Theor.*, 55(7):3346–3359, jul 2009. ISSN 0018-9448. doi: 10.1109/TIT.2009.2021311. URL https://doi.org/10.1109/TIT.2009.2021311.

Solomon Kullback. *Information Theory and Statistics*. Wiley, 1959.

Xingguo Li, Tuo Zhao, Raman Arora, Han Liu, and Mingyi Hong. On faster convergence of cyclic block coordinate descent-type methods for strongly convex minimization. *Journal of Machine Learning Research*, 18(184):1–24, 2018.

Song Liu, Makoto Yamada, Nigel Collier, and Masashi Sugiyama. Change-point detection in time-series data by relative density-ratio estimation. *Neural Networks*, 43:72–83, 2013.

Y. Mei. Information bounds and quickest change detection in decentralized decision systems. *IEEE Transactions on Information Theory*, 51 (7):2669–2681, 2005. doi: 10.1109/TIT.2005.850159.

Y. MEI. Efficient scalable schemes for monitoring a large number of data streams. *Biometrika*, 97(2):419–433, 2010. ISSN 00063444,

References II

S. Aminikhanghahi and D. Cook. A survey of methods for time series change point detection. *Knowledge and Information Systems*, 51: 339–367, 2016.

Imre Csiszár. On topological properties of f-divergences. *Studia Scientiarum Mathematicarum Hungarica*, 2:329–339, 1967.

Alejandro de la Concha, Argyris Kalogeratos, and Nicolas Vayatis. Collaborative likelihood-ratio estimation over graphs, 2022.

Alejandro de la Concha, Argyris Kalogeratos, and Nicolas Vayatis. Online centralized non-parametric change-point detection via graph-based likelihood-ratio estimation, 2023. URL https://arxiv.org/abs/2301.03011.

Alejandro de la Concha Duarte, Nicolas Vayatis, and Argyris Kalogeratos. Offline detection of change-points in the mean for stationary graph signals. In *Proceedings of International Conference on Artificial Intelligence and Statistics*, volume 130, pages 3430–3438, 2021.

A. Ferrari and C. Richard. Non-parametric community change-points detection in streaming graph signals. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5545–5549, 2020.

Olympia Hadjiliadis, Hongzhong Zhang, and H. Vincent Poor. One shot schemes for decentralized quickest change detection. *IEEE Trans. Inf. Theor.*, 55(7):3346–3359, jul 2009. ISSN 0018-9448. doi: 10.1109/TIT.2009.2021311. URL https://doi.org/10.1109/TIT.2009.2021311.

Solomon Kullback. *Information Theory and Statistics*. Wiley, 1959.

Xingguo Li, Tuo Zhao, Raman Arora, Han Liu, and Mingyi Hong. On faster convergence of cyclic block coordinate descent-type methods for strongly convex minimization. *Journal of Machine Learning Research*, 18(184):1–24, 2018.

Song Liu, Makoto Yamada, Nigel Collier, and Masashi Sugiyama. Change-point detection in time-series data by relative density-ratio estimation. *Neural Networks*, 43:72–83, 2013.

Y. Mei. Information bounds and quickest change detection in decentralized decision systems. *IEEE Transactions on Information Theory*, 51 (7):2669–2681, 2005. doi: 10.1109/TIT.2005.850159.

Y. MEI. Efficient scalable schemes for monitoring a large number of data streams. *Biometrika*, 97(2):419–433, 2010. ISSN 00063444,