At Most One Change Offline: Essential Statistical and Computational Concepts

Arnaud Liehrmann & Guillem Rigaill

Sorbonne Université & INRAE

February 25, 2025

Plan

Problem Set-up

- Single changepoint detection assumes at most one change (AMOC) in the underlying probability distribution generating the data $Y_{1:n}$.
- Hypothesis testing problem:
  - **(H$_0$)**: No changepoint in $Y_{1:n}$.
  - **(H$_1$)**: One changepoint at some unknown time

$$\tau \in \{1, \ldots, n-1\}$$

.

- If **(H$_1$)** is true, estimate the location of the change τ.

A first fairly generic solution

- Construct/Use a test statistic
  - Comparing the parameter before and after the change at τ
  - Ex: using a likelihood ratio to compare the likelihood of the data with and without a change at τ.
- Apply it to **all possible changepoints**
- Examples of assumptions:
  - i.i.d Gaussian with known variance.
  - i.i.d Gaussian with known mean.
  - i.i.d Gaussian with unknown variance.
  - i.i.d Poisson, Exponential, etc.
  - i.i.d Non-Parametric model
  - ...

An example: i.i.d Gaussian in Mean

- Data $Y_{1:n}$ are univariate i.i.d. Gaussian with a potential change in the mean parameter μ_t.
- Formally:

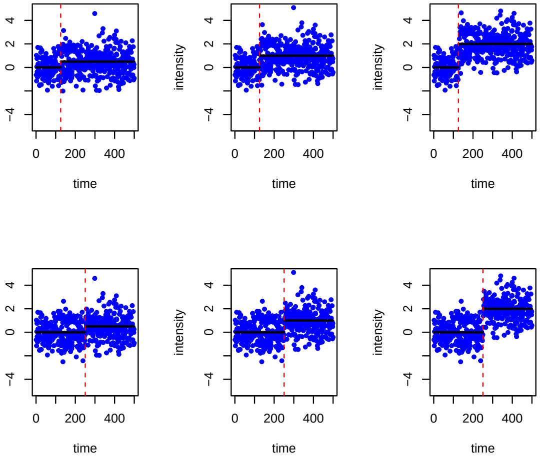
$$Y_t \sim \mathcal{N}(\mu_t, \sigma^2), \quad t = 1, \cdots, n$$

- Under $\mathbf{H}_1$, there is one changepoint at some unknown time

$$\tau \in \{1, \ldots, n-1\}$$

.

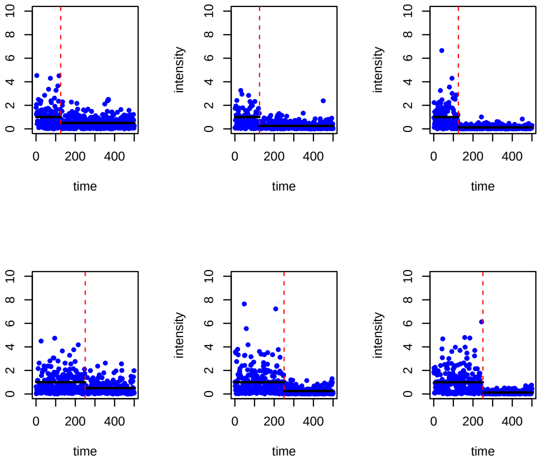
Gaussian Model Illustration



An example: i.i.d Exponential

- Data $Y_{1:n}$ are i.i.d. Exponential with a potential change in the parameter λ_t.
- Formally:
$$Y_t \sim \text{Exp}(\lambda_t), \quad t = 1, \cdots, n$$
- Under $\mathbf{H}_1$, there is one changepoint at some unknown time $\tau \in \{1, \ldots, n-1\}$.

Exponential Model Illustration



Stop and Think

Stop and Think

Visualizing the previous plots, how do you think the difficulty of detecting a changepoint depends on the height of the change and the position of the changepoint?

Exercise

Exercise

Implement in Python, a function that simulates i.i.d. Gaussian data with a single change in the mean. Then test this implementation with various values of n, τ, μ_1, μ_2, and σ. After implementing this pseudocode and obtaining the simulated data, you can plot:

- the mean vector as a line or step plot;
- a scatter plot of the observed data;
- add a vertical line at the change-point τ.

Plan

Maximum Likelihood Inference

- For an i.i.d. parametric model, arguably the vanilla approach it to use maximum likelihood and derive a likelihood ratio test.
- Under $\mathbf{H}_0$, the maximum likelihood is:

$$\max_{\theta} \prod_{t=1}^{n} f_{Y_t}(\theta).$$

- Under $\mathbf{H}_1$, the maximum likelihood is:

$$\max_{\theta_1,\theta_2} [(\prod_{t=1}^{\tau} f_{Y_t}(\theta_1))(\prod_{t=\tau+1}^{n} f_{Y_t}(\theta_2))]$$

Likelihood Ratio Test (LRT)

- The LRT statistic at τ is:

$$LR_\tau = -2\log\left\{\frac{\max_\theta \prod_{t=1}^{n} f_{Y_t}(\theta)}{\max_{\theta_1,\theta_2}[(\prod_{t=1}^{\tau} f_{Y_t}(\theta_1))(\prod_{t=\tau+1}^{n} f_{Y_t}(\theta_2))]}\right\}$$

- Often expressed as a difference between the log-likelihoods:

$$\begin{aligned} LR_\tau = &-2\max_\theta \sum_{t=1}^{n} \log(f_{Y_t}(\theta)) \\ &+ 2\max_{\theta_1,\theta_2}[\sum_{t=1}^{\tau} \log(f_{Y_t}(\theta_1)) + \sum_{t=\tau+1}^{n} \log(f_{Y_t}(\theta_2))]. \end{aligned}$$

Maximum LRT Statistic

- Consider all possible τ:

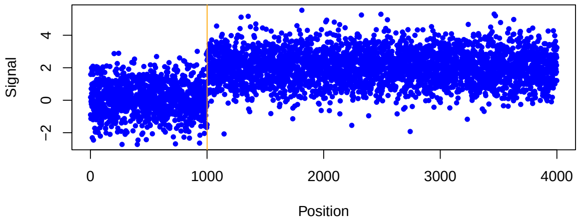
$$LR_{\max} = \max_{\tau \in \{1, \cdots, n-1\}} LR_{\tau}.$$

- If $LR_{\max}$ is large enough, declare a change and estimate its position as:

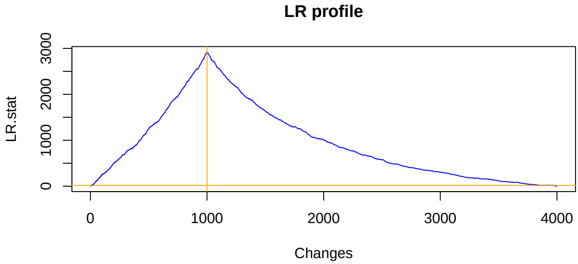
$$\hat{\tau} = \arg \max_{\tau \in \{1, \cdots, n-1\}} LR_{\tau}.$$

- We need to set a threshold β...

In practise



Signal
4
2
0
−2
0
1000
2000
3000
4000
Position



LR profile
LR.stat
3000
2000
1000
0
0
1000
2000
3000
4000
Changes

Calibrating the Threshold

- The choice of the threshold β is crucial.
- It should be
  - large enough to avoid false positives
  - small enough to detect true changes
- And ideally, it should provide some robustness to model error/mispecification.

Setting an "Appropriate" Threshold

- For some generic models, mathematically deriving an optimal threshold is an open question (to the best of our knowledge)
- Assuming i.i.d Gaussian errors, some precise answers exist based on asymptotic or non-asymptotic arguments [Hinkley, 1971, Gombay and Horvath, 1990, Verzelen et al., 2023]
- These theoretical results are very important as they provide insight into the difficulties of the problem and how we could overcome some of them but typically do not provide explicit thresholds (need to calibrate certain constants...)

Setting the Threshold Using Monte Carlo

- Use Monte Carlo to approximate the null distribution of test statistics.
- Easily applied to more complicated change-point scenarios (e.g. complex distribution, constraints between parameters...)
- Drawback: Threshold depends on the size of the data, requiring re-run simulations for new data sizes.

Setting the Threshold with a Bonferroni Correction

- Control the family-wise error rate using a Bonferroni bound.
- Informally, set the per p-value threshold at α/n.

Bonferroni Correction Proof (1)

- Assume we can provide a value $\beta_\tau(\alpha)$ such that

$$P(LR_\tau \geq \beta_\tau(\alpha)) \leq \alpha.$$

- Assuming $\beta_\tau(\alpha) = \beta(\alpha)$ does not depends on τ

Bonferroni Correction Proof (2)

- We control the probability that $LR_{\max}$ exceeds $\beta(\alpha/(n-1))$ using a union bound:

$$
\begin{aligned}
P\left(LR_\tau \geq \beta\left(\frac{\alpha}{n-1}\right)\right) &= P\left(\max_{\tau\in\{1,\cdots,n-1\}} LR_\tau \geq \beta\left(\frac{\alpha}{n-1}\right)\right) \\
&= P\left(\bigcup_{\tau\in\{1,\cdots,n-1\}} \left(LR_\tau \geq \beta\left(\frac{\alpha}{n-1}\right)\right)\right) \\
&\leq \sum_{\tau=1}^{n-1} P\left(LR_\tau \geq \beta\left(\frac{\alpha}{n-1}\right)\right) \\
&\leq \sum_{\tau=1}^{n-1} \frac{\alpha}{n-1} = \alpha
\end{aligned}
$$

Exercise

Exercise

1. Using the previous Bonferonni argument construct a procedure to detect a change in the mean (for a known variance).
2. Test it on datasets of size $n = 2^{10}$. Check that is controlling H_0. Assess its power to detect changes at positions 2^i, for $i = 1, 2, ..., 9$.
3. What is the computational complexity of your approach?
4. Propose a Monte Carlo-based calibration of your procedure. Compare its power to the Bonferonni version.
5. Test the two approaches for errors simulated as a student t statistic. Explore for various values of the degree of freedom.

Plan

LRT for i.i.d Gaussian with known variance

- We now focus on the Gaussian change in mean model with independent errors.
- We will
  - Relate the LRT statistic to the Cusum statistic
  - Propose an efficient computation of the LRT (or Cusum) statistic
  - Re-use the Bonferonni calibration of the threshold to study the detection power

LRT Statistic Derivation

We consider the data

$$
\begin{aligned}
y_t &= \mu_1 + \varepsilon_t \qquad \text{for } t \leq \tau \\
y_t &= \mu_2 + \varepsilon_t \qquad \text{for } t > \tau \\
&\text{with} \quad \varepsilon_t \sim \mathcal{N}(0, \sigma^2)
\end{aligned}
$$

LRT Statistic Derivation

- and we get

$$
\begin{aligned}
LR_\tau = & -2 \min_{\mu} \left(-\frac{1}{2\sigma^2} \sum_{t=1}^{n} (y_t - \mu)^2 \right) \\
& + 2 \min_{\mu_1, \mu_2} \left(-\frac{1}{2\sigma^2} \left(\sum_{t=1}^{\tau} (y_t - \mu_1)^2 + \sum_{t=\tau+1}^{n} (y_t - \mu_2)^2 \right) \right) \\
& + \quad \tau \log(2\pi\sigma^2) + (n - \tau) \log(2\pi\sigma^2) - n \log(2\pi\sigma^2).
\end{aligned}
$$

Simplification of LRT Statistic

- Up to some constants this simplifies to

$$LR_\tau = \frac{1}{\sigma^2} \min_{\mu} \sum_{t=1}^{n} (y_t - \mu)^2 - \frac{1}{\sigma^2} \min_{\mu_1, \mu_2} \left(\sum_{t=1}^{\tau} (y_t - \mu_1)^2 + \sum_{t=\tau+1}^{n} (y_t - \mu_2)^2 \right).$$

Empirical Means Substitution

- Optimal values are obtained considering the empirical mean of each segment $1 : \tau$, $\tau + 1 : n$ and $1 : n$

$$LR_\tau = \frac{1}{\sigma^2}\left[\sum_{t=1}^{n}(y_t - \bar{y}_{1:n})^2 - \sum_{t=1}^{\tau}(y_t - \bar{y}_{1:\tau})^2 - \sum_{t=\tau+1}^{n}(y_t - \bar{y}_{(\tau+1):n})^2\right].$$

$$\bar{y}_{u:l} = \frac{1}{l - u + 1}\sum_{t=u}^{l} y_t.$$

Exercise

Exercise

1. Do the same calculation for a Gaussian change in mean model with unknown variance. Compare the $\arg\max$ with the one obtained for a known variance.
2. (AT HOME) Do the same calculation for a Gaussian change in variance only.

Outline

Cusum-like Reformulation

- CUSUM statistics, introduced by Page [Page, 1954], detect changes in an online setting.
- Compare the likelihood of a model without a change and a model with a change.
- In part for computational efficiency, θ_1 and θ_2 are assumed known.

CUSUM Statistic

$$Cusum_n(\theta_1, \theta_2) = \max_{\tau < n} \left\{ \sum_{t=\tau+1}^{n} \log \left(\frac{f_{Y_t}(\theta_2)}{f_{Y_t}(\theta_1)} \right) \right\}$$

- This idea has led to numerous developments for changepoints [Aue and Kirch, 2024].
- For a Gaussian model, the CUSUM statistic can be rewritten as the likelihood ratio statistic assuming we would optimize the value of θ_1 and θ_2.

CUSUM Statistic for Gaussian Model

$$C_\tau = \sqrt{\frac{\tau(n-\tau)}{n}} \left(\bar{y}_{1:\tau} - \bar{y}_{(\tau+1):n}\right)$$

- If no changepoint exists at τ, the prefactor ensures the statistic follows a normal random variable with variance σ^2.

Detection of Changepoint

$$LR_{\max} = C_{\max}^2 = \max_{\tau \in \{1, \cdots, n-1\}} \frac{C_\tau^2}{\sigma^2} > \beta$$

- Declare a change at τ if the squared and variance-normalized CUSUM statistic exceeds a threshold β.

Exercise

Prove that the CUSUM statistic can be rewritten as the likelihood ratio statistic.

Proof 1 (not too slow)

- Consider the left terms with t from 1 to τ.

$$
\begin{aligned}
A_{left} &= \sum_{1}^{\tau}(y_t - \bar{y}_{1:\tau})^2 - \sum_{1}^{\tau}(y_t - \bar{y}_{1:n})^2 \\
&= \sum_{1}^{\tau}(2y_t - \bar{y}_{1:\tau} - \bar{y}_{1:n})(\bar{y}_{1:n} - \bar{y}_{1:\tau}) \\
&= \sum_{1}^{\tau}(y_t - \bar{y}_{1:n})(\bar{y}_{1:n} - \bar{y}_{1:\tau}) \\
&= \tau(\bar{y}_{1:n} - \bar{y}_{1:\tau})^2
\end{aligned}
$$

Proof 1 (continued)

$$\begin{aligned}
\bar{y}_{1:n} - \bar{y}_{1:\tau} &= \frac{\tau \sum_1^n y_t - n \sum_1^\tau y_t}{n\tau} \\
&= \frac{\tau \sum_{\tau+1}^n y_t - (n-\tau) \sum_1^\tau y_t}{n\tau} \\
&= \frac{n-\tau}{n} (\bar{y}_{(\tau+1):n} - \bar{y}_{1:\tau})
\end{aligned}$$

$$A_{left} = \frac{\tau(n-\tau)^2}{n^2} (\bar{y}_{(\tau+1):n} - \bar{y}_{1:\tau})^2$$

- Then recover A_{right} by symmetry and combine.

Outline

Computation

- One critical aspect of modern (multi)-changepoint approaches is their runtime complexity.
- You want the approach to be both statistically efficient and computationally scalable.
- The offline AMOC is rather simple, but a good opportunity to present a simple yet key trick.

Stop and Think

Stop and Think

What is the computational complexity of a naive implementation of the CUSUM statistic when computed iteratively along a time series?

An Example

- Let us compute the CUSUM for the vector:

$$y_{1:4} = (0.8, 1.2, 4.5, 4.3),$$

- Assuming the observations are Gaussian with $\sigma^2 = 1$.
- The possible changepoint positions are:

$$\tau \in \{1, 2, 3\}.$$

Empirical Means

- Before computing the CUSUM statistics for each potential changepoint, we first need to determine the empirical means for each segment.
- Specifically, for each τ, we compute the means of the subsequences:

$$\bar{y}_{1:\tau} = \frac{1}{\tau} \sum_{i=1}^{\tau} y_i, \quad \bar{y}_{(\tau+1):4} = \frac{1}{4-\tau} \sum_{i=\tau+1}^{4} y_i.$$

Computing Means

$$\begin{aligned} \bar{y}_{1:1} &= y_1 = 0.8, & \bar{y}_{2:4} &= \frac{1.2 + 4.5 + 4.3}{3} = 3.33, \\ \bar{y}_{1:2} &= \frac{0.8 + 1.2}{2} = 1.0, & \bar{y}_{3:4} &= \frac{4.5 + 4.3}{2} = 4.4, \\ \bar{y}_{1:3} &= \frac{0.8 + 1.2 + 4.5}{3} = 2.17, & \bar{y}_{4:4} &= y_4 = 4.3. \end{aligned}$$

CUSUM Statistic

- The squared CUSUM statistic for each τ is then given by:

$$C_\tau^2 = \frac{\tau(4-\tau)}{4}(\bar{y}_{1:\tau} - \bar{y}_{(\tau+1):4})^2.$$

Computing CUSUM Values

$$
\begin{aligned}
C_1^2 &= \frac{3 \times 1}{4}(0.8 - 3.33)^2 = 0.75 \times 6.4 = 4.8, \\
C_2^2 &= \frac{2 \times 2}{4}(1.0 - 4.4)^2 = 11.56, \\
C_3^2 &= \frac{1 \times 3}{4}(2.17 - 4.3)^2 = 0.75 \times 4.53 = 3.4.
\end{aligned}
$$

Maximum CUSUM Statistic

- Thus, the maximum CUSUM statistic is:

$$C^2_{\mathrm{max}} = 11.56 \text{ at } \tau = 2.$$

- Recall, that to call a changepoint, we would compare C^2_{max} to a threshold value β.
- If $C^2_{\mathrm{max}} > \beta$, we conclude that there is a change

Naive Implementation

- A naive implementation of the CUSUM statistic computes the means $\bar{y}_{1:\tau}$ and $\bar{y}_{\tau+1:n}$ for each τ in $\mathcal{O}(n)$.
- This leads to an overall computational complexity of $\mathcal{O}(n^2)$.

Efficient Implementation

- We can be much faster by sequentially computing partial sums:
$$S_t = \sum_{t=1}^{n} y_t.$$
- By computing partial sums incrementally, we avoid redundant calculations.
- This achieves a linear time complexity of $\mathcal{O}(n)$, making the method scalable for large datasets.

Efficient CUSUM Algorithm I

Algorithm 1 Efficient CUSUM Algorithm

Require: Time series $y = (y_1, \cdots, y_n)$, threshold c, variance σ^2
Ensure: Changepoint estimate $\hat{\tau}$, maximum CUSUM statistic $C^2_{\max}$

$n \leftarrow$ length of y
$C^2_{\max} \leftarrow 0$
$\hat{\tau} \leftarrow 0$
$S_n \leftarrow \sum_{t=1}^{n} y_t$
$S \leftarrow 0$
for $t = 1$ to $n - 1$ **do**
    $S \leftarrow S + y_t$
    $\bar{y}_{1:t} \leftarrow \frac{S}{t}$
    $\bar{y}_{(t+1):n} \leftarrow \frac{S_n - S}{n - t}$
    $C^2_t \leftarrow \frac{t(n-t)}{n}(y_{1:t} - \bar{y}_{(t+1):n})^2$
    if $C^2_t > C^2_{\max}$ **then**
        $C^2_{\max} \leftarrow C^2_t$
        $\hat{\tau} \leftarrow t$
    end if
end for
if $\frac{C^2_{\max}}{\sigma^2} > c$ **then**
    return $\hat{\tau}, C^2_{\max}$ ▷ Changepoint detected
else
    return NULL, $C^2_{\max}$ ▷ No changepoint detected
end if

Incremental Update of the Mean

- This incremental update of the mean is key to the efficiency of many algorithms and methods for multiple changepoint detection.
- Particularly in approaches based on dynamic programming [Auger and Lawrence, 1989, Killick et al., 2012, Rigaill, 2015] and approaches based on local optimization or isolation [Fryzlewicz, 2014, Fryzlewicz, 2020, Kovács et al., 2023, Anastasiou and Fryzlewicz, 2022]
- This incremental approach works for many other distributions than just the Gaussian.
- However, it does not always work.

Exercise

Exercise

Give an example of a distribution, or model for which the cumsum trick does not work.

Exercise

Exercise

Implement the *efficient CUSUM Algorithm* in Python. Your function should:

1. **Return:**
   - The estimated changepoint, $\hat{\tau}$.
   - The maximum CUSUM statistic, $C^2_{\max}$.
   - The full sequence $\{C^2_\tau\}$ for $\tau = 1, \dots, n-1$.
2. **Test** your implementation on multiple synthetic datasets generated by the function developed in *Exercise 1*.

Outline

Controlling the CUSUM Statistics

We aim to find a threshold for c such that we declare a changepoint if

$$\max_{\tau < n} C_\tau > \beta$$

A Bonferroni-like Bound

Assuming the y_t are all Gaussian with $\sigma^2 = 1$ (without loss of generality), the C_τ are $\mathcal{N}(0, 1)$ and we can use a sub-Gaussian concentration bound.

$$\begin{aligned} P\left(\max_{\tau<n} |C_\tau| > \beta\right) &= P\left(\bigcup_{\tau<n} |C_\tau| > \beta\right) \\ &\leq \sum_{\tau<n} P\left(|C_\tau| > \beta\right) \\ &\leq \sum_{\tau<n} 2\exp\left(-\frac{\beta^2}{2}\right) \end{aligned}$$

Threshold Calculation

Taking $\beta^2 = 2\log(n-1) - 2\log(\alpha)$, we get

$$P\left(\max_{\tau<n} |C_\tau| > \beta\right) \leq 2\alpha$$

Detection power

Consider a change of size δ at τ^*:

$$C_{\tau^*} = \sqrt{\frac{\tau^*(n-\tau^*)}{n}}(\delta + \bar{\varepsilon}_{\tau^*+1:n} - \bar{\varepsilon}_{1:\tau^*})$$

- We now aim to estimate the power we get from our crude union-bound threshold

Cusum statistic for the true changepoint C_{τ^*}

We consider the data

$$\begin{aligned} y_t &= 0 + \varepsilon_t \qquad \text{for } t \leq \tau^* \\ y_t &= \delta + \varepsilon_t \qquad \text{for } t > \tau^* \\ &\text{with} \quad \varepsilon_t \sim \mathcal{N}(0, \sigma^2) \end{aligned}$$

Rewriting C_{τ^*} we get:

$$C_{\tau^*} \quad = \quad \delta\sqrt{\frac{\tau^*(n-\tau^*)}{n}} + \sqrt{\frac{\tau^*(n-\tau^*)}{n}}(\bar{\varepsilon}_{\tau^*:n} - \bar{\varepsilon}_{1:\tau^*})$$

Using Sub-Gaussian Concentration Again

- With probability at least $1 - 2\exp(-0.5x^2)$ we have:

$$\left| \sqrt{\frac{\tau^*(n-\tau^*)}{n}} (\bar{\varepsilon}_{\tau^*:n} - \bar{\varepsilon}_{1:\tau^*}) \right| \leq x$$

Condition for Change Detection

Considering $x = \sqrt{-2\log(\alpha)}$, we get that if

$$\delta\sqrt{\frac{\tau(n-\tau)}{n}} > \sqrt{2\log((n-1)/\alpha)} + \sqrt{-2\log(\alpha)}$$

then with probability at least $1-\alpha$, C_{τ^*} would pass the threshold and we would detect a change.

Key Insights

From this, we get two important pieces of information:

1. The power depends on the position of the change, with changes close to the border being harder to detect.
2. Consider τ to be proportional to n, $\tau = a \cdot n$ (with $a \in (0, 1)$), we detect the change if

$$\delta\sqrt{a(1-a)n} \geq \sqrt{2\log((n-1)/\alpha)} + \sqrt{-2\log(\alpha)}.$$

- this will always happen eventually as $\sqrt{n}$ grows faster than $\log(n)$.

Stop and Think

Stop and Think

Using the previous power calculation (assuming a variance of 1), plot the minimum jump size you could detect as a function of n and then as a function of τ^* for $n = 10^3, 10^4, 10^5$. Repeat using a $\sqrt{2 \log \log(n)}$ threshold. What do you think?

Plan

Exercise

Exercise

1. Check the distribution of $\hat{\tau}$ returned by the CUSUM procedure with the Bonferroni-like bound on simulated datasets of size $n = 100$ under H_0.
2. Assess the power of the procedure to detect changes at positions 2^i, for $i = 1, 2, \ldots, 9$, on profiles of size $n = 100$ under H_1. What do you notice?
3. Does the distance between the true and estimated changepoints vary with the position of the true changepoint?

Exercise

Exercise

1. In this exercise, we will consider a restriction on the size of the segments (say at least 10 data-points). Propose a Monte Carlo-based calibration of this new procedure.
2. Test its H_0 control and power on data simulated with i.i.d Gaussian errors, then with i.i.d. Student's t-distributions with degrees of freedom 3.
3. Compare to the results you had without the restriction.

Exercise

Exercise (AT HOME)

1. In this exercise, we will consider the case where the variance is unknown and explore two strategies.
2. In the first strategy, we use the statistic defined earlier in *Exercise 6* and calibrate it using Monte Carlo simulation.
3. In the second strategy, we pre-estimate the variance using the Median Absolute Deviation (MAD) estimator:

$$\mathrm{MAD} = \mathsf{median}\Big(\big|Y_i - \mathsf{median}(Y)\big|\Big).$$

4. For data drawn from a normal distribution, the MAD can be scaled to provide a consistent estimator for the standard deviation:

$$\hat{\sigma} = 1.4826 \times \mathrm{MAD}.$$

Exercise

Exercise (AT HOME)

This exercise is for the more mathematically inclined and can be skipped. It should give some intuition as to why $\sqrt{\log(\log(n))}$ from [Gombay and Horvath, 1990, Verzelen et al., 2023] is feasible.

1. In [Verzelen et al., 2023], the following lemma (see next slide) is proven. Use this lemma to show that a threshold of order $\sqrt{\log(\log(n))}$ does control H_0.
2. Assume you only want to test changepoints that are sufficiently far from the borders (say in $(an : (1-a)n)$ for $0 < a < 1$. Does it change your bound?

Lemma

The following lemma is found in [Verzelen et al., 2023]

Lemma

Let $\epsilon_1, \ldots, \epsilon_n$ be independent centered sub-Gaussian random variables such that

$$E[e^{s\epsilon_i}] \leq e^{\frac{s^2}{2}}, \quad \text{for any } i \geq 1 \text{ and any } s > 0.$$

Then, for any integer $d > 0$, any $\alpha > 0$, and any $x > 0$,

$$P\left[\max_{k \in [d,(1+\alpha)d]} \sum_{i=1}^{k} \epsilon_i \frac{1}{\sqrt{k}} \geq x\right] \leq \exp\left(-\frac{x^2}{2(1+\alpha)}\right).$$

Plan

Conclusion

We conclude this chapter on the detection of a single change here a

- we do not further discuss the localization error of the CUSUM procedure or the construction of a confidence interval for the change
  - Some/Many proposals have been made, their derivations are typically technical and outside the scope of this class
  - An arguably natural way to address this question and cope with the discrete nature of the changepoint is to adopt a Bayesian framework and construct credibility intervals.

Anastasiou, A. and Fryzlewicz, P. (2022).
Detecting multiple generalized change-points by isolating single ones.
Metrika, 85(2):141–174.

Aue, A. and Kirch, C. (2024).
The state of cumulative sum sequential changepoint testing 70 years after page.
Biometrika, 111(2):367–391.

Auger, I. E. and Lawrence, C. E. (1989).
Algorithms for the optimal identification of segment neighborhoods.
Bulletin of mathematical biology, 51(1):39–54.

Fryzlewicz, P. (2014).
Wild binary segmentation for multiple change-point detection.

Fryzlewicz, P. (2020).
Detecting possibly frequent change-points: Wild binary segmentation 2 and steepest-drop model selection.

Journal of the Korean Statistical Society, 49(4):1027–1070.

Gombay, E. and Horvath, L. (1990).
Asymptotic distributions of maximum likelihood tests for change in the mean.
Biometrika, 77(2):411–414.

Hinkley, D. V. (1971).
Inference about the change-point from cumulative sum tests.
Biometrika, 58(3):509–523.

Killick, R., Fearnhead, P., and Eckley, I. A. (2012).
Optimal detection of changepoints with a linear computational cost.
Journal of the American Statistical Association, 107(500):1590–1598.

Kovács, S., Bühlmann, P., Li, H., and Munk, A. (2023).
Seeded binary segmentation: a general methodology for fast and optimal changepoint detection.
Biometrika, 110(1):249–256.

Page, E. S. (1954).
Continuous inspection schemes.
Biometrika, 41(1/2):100–115.

Rigaill, G. (2015).
A pruned dynamic programming algorithm to recover the best segmentations with 1 to $k\_\{max\}$ change-points.
Journal de la Société Française de Statistique, 156(4):180–205.

Verzelen, N., Fromont, M., Lerasle, M., and Reynaud-Bouret, P. (2023).
Optimal change-point detection and localization.
The Annals of Statistics, 51(4):1586–1610.