Changepoint detection in practice

15-th International Workshop on Operation Research

Clément Laroche

June 12, 2023

Université Paris 1 - Panthéon Sorbonne

Outline

1. Brief reminder
2. Cost function
3. Search methods

Brief reminder

Changepoint detection

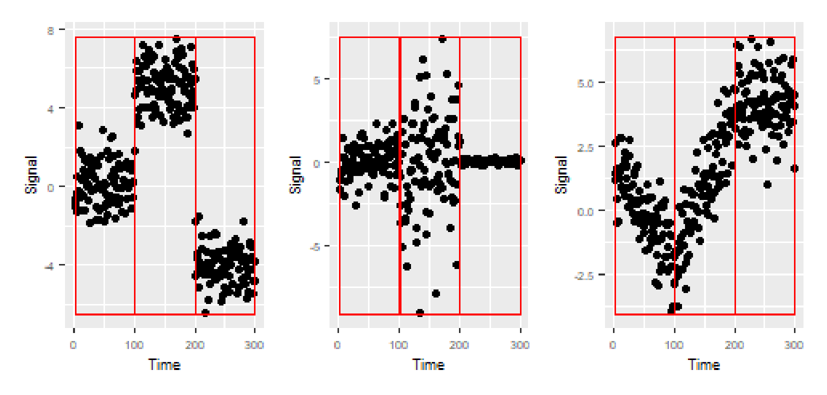

Figure 1: Examples of changepoint in signals in mean (left), variance (middle) and trend (right).

Online vs offline methods

- Online methods aim to detect changes as soon as they occur in a real-time setting.
- Offline methods aim to detect changes retrospectively when all samples are received.

This lesson focuses on offline methods only.

Problem statement

Notations:

- $\boldsymbol{y} = \{y_1, \ldots, y_n\}$ random non-stationnary process.
- K^* the number of change point in $\boldsymbol{y}$.
- $t_1^* \leq \ldots \leq t_{K^*}^*$ the indices where the K^* change points are located

Change point detection problem : choosing the best segmentation $\mathcal{T} = \{t_1, \ldots, t_K\}$ according to a quantitative criterion $V(\mathcal{T})$ that must be minimized.

Problem statement

Problem 1: the number of changes K is fixed and known. Then, we try to minimize the following quantity:

$$\min_{|\mathcal{T}|=K} V(\mathcal{T})$$

Problem 2: the number of changes K is unknwon. Then, the problem becomes:

$$\min_{\mathcal{T}} V(\mathcal{T}) + \mathrm{pen}(\mathcal{T})$$

where $\mathrm{pen}(\mathcal{T})$ is an appropriate measure of the complexity of a segmentation.

Structure of change point detection methods

Change point detection methods are composed of three elements:

- the cost function that will define the quantitative criterion $V(\mathcal{T})$.
- the search method that will compute the solution to problem 1 or 2.
- the constraint that will be only used in Problem 2 and that allows the estimation of the number of changes K^*.

Typology given in Truong (2018).

Cost function

Cost function definition

For a given segmentation $\mathcal{T} = \{t_1, \ldots, t_K\}$, we suppose V can be written as:

$$V(\mathcal{T}) = \sum_{k=0}^{K} c(y_{t_k:t_{k+1}})$$

where $c()$ is the cost function.

Summary:

- Low values of $c()$ occurs when the data in $y_{t_k:t_{k+1}}$ is homogeneous.
- High values of $c()$ occurs when the data in $y_{t_k:t_{k+1}}$ is heterogeneous.

Cost function example

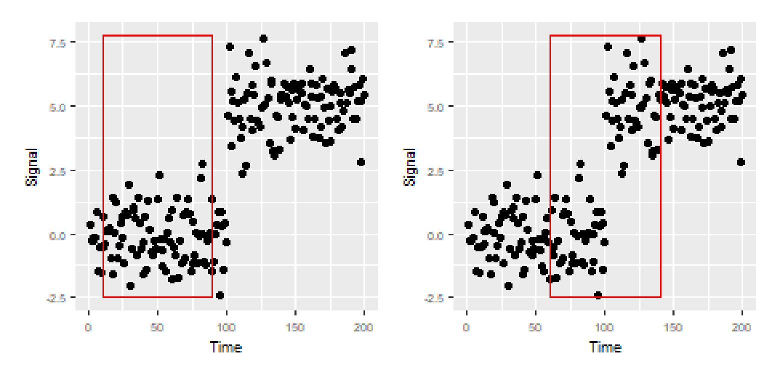

Figure 2: If $c()$ is set as the negative log likelihood of the signal, the segment on the left figure will have a low cost, the segment on the right an elevated cost.

Simulated data set

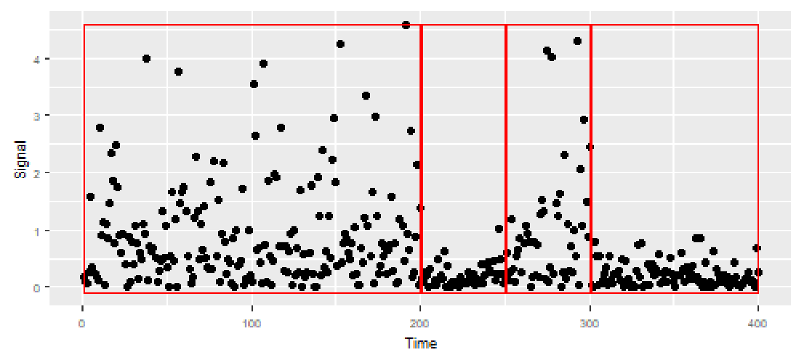

Figure 3: We will work on simulated signals composed of 400 realisations of exponential distribution of parameter λ. Changes in the scale parameter are positionned in position 200,250 and 300. The scale parameter of each segment are (1,5,1,4).

Resulting cost function

For a given segment $y_{a:b}$ of size m, the negative log-likelihood can be written as :

$$c(y_{a:b}, \lambda) = -m \log(\lambda) + \lambda \sum_{i=a+1}^{b} y_i$$

Cost function choice :

$$c(y_{a:b}) = -m \log(\lambda_{MLE}) + \lambda_{MLE} \sum_{i=a+1}^{b} y_i$$

with $\lambda_{MLE} = \frac{1}{\sum_{i=a+1}^{b} y_i}$ the maximum likelihood estimator of λ for the segment $y_{a:b}$.

Search methods

Two types of search methods

1. Exact search methods (Optimal partition, PELT)
2. Approximate methods (window sliding, binary segmentation,bottom-up segmentation...)

Solution to problem 1: Optimal Partition

Here the number of changepoints K is fixed.

$$
\begin{aligned}
\min_{|\mathcal{T}|=K} V_{y_{1:n}}(\mathcal{T}) &= \min_{0=t_0<t_1<\cdots<t_K<t_{K+1}=n} \sum_{k=0}^{K} c(y_{t_k+1:t_{k+1}}) \\
&= \min_{t\leq n-K} \{ c(y_{1:t}) + \min_{t=t_0<t_1<\cdots<t_K<t_{K+1}=n} \sum_{k=0}^{K} c(y_{t_k+1:t_{k+1}}) \} \\
&= \min_{t\leq n-K} \{ c(y_{1:t}) + \min_{|\mathcal{T}|=K-1} V_{y_{t:n}}(\mathcal{T}) \}
\end{aligned}
$$

Solution based on dynamic programming.

Solution to problem 1: Optimal Partition i

input : signal $y_{1:n}$, cost function $c()$, number of changepoints $K \geq 1$
Create M_1 a $n \times n$ empty matrix
for all (u, v) such that $1 \leq u < v \leq n$ **do**
    $M(u, v) \leftarrow c(y_{u:v})$
end for
if $K + 1 > 2$ **then**
    for $k = 2, ..., K$ **do**
        for all $u, v \in \{1, .., n\}$ such that $v - u > k$ **do**
            $M_k(u, v) \leftarrow \min_{u+k-1 \leq t < v} M_{k-1}(u, t) + M_1(t + 1, v)$
        end for
    end for
end if
$L \leftarrow (0, ..., 0)$ vector of size $K + 1$
$L(K + 1) \leftarrow n$
$k \leftarrow K + 1$

Solution to problem 1: Optimal Partition ii

while $k > 1$ **do**

$s \leftarrow L(k)$

$t^* \leftarrow \arg\min_{k-1 \leq t < s} M_{k-1}(1, t) + M_1(t+1, s)$

$L(k-1) \leftarrow t^*$

$k \leftarrow k - 1$

end while

Output: a list L of K estimated changepoints (with n as a last coordinate).

The problem is, in practive, the number of change points is not known before hand. In that case, we are not in the configuration of problem 1 anymore.

Problem 2: introducing linear penalty

- Problem 2 is written as:

$$\min_{\mathcal{T}} V(\mathcal{T}) + \mathrm{pen}(\mathcal{T})$$

- Linear penalty means that:

$$\mathrm{pen}(\mathcal{T}) = `\beta|\mathcal{T}|$$

where β is the penalty weight.

Solution to Problem 2: PELT

Pruned Exact Linear Time algorithm Killick, Fearnhead, and Eckley (2012) is also based on dynamic programming.

Notation: $F(t) = \min_{\mathcal{T}}\{V_{y_{1:t}}(\mathcal{T}) + \beta|\mathcal{T}|\}$

PELT introduces a pruning rule to discard potential change points, that can be written as: for all $t < s < n$, if

$$F(t) + c(y_{t+1:s}) + \beta \geq F(s)$$

holds, then t can never be the last change point prior to s.

Solution to Problem 2: PELT

input : the data $y_1, ..., y_n$, a cost function $c()$, the penalty term β and a minimal segment length n_{min}

initialisations : F a vector of size n, $R_1 = \{0\}$, $CP(0) = NULL$
$F(i) = -\beta$, for all $i \in \{1, ..., n_{min}\}$
for all $\tilde{t} = n_{min} + 1, ..., n$ **do** :
    Compute $F(\tilde{t}) = \min_{t \in R_{\tilde{t}} || t - \tilde{t}| \geq n_{min}} \{F(t) + c(y_{(t+1):\tilde{t}}) + \beta\}$
    Compute $\bar{t} = \arg\min_{t \in R_{\tilde{t}} || t - \tilde{t}| \geq n_{min}} \{F(t) + c(y_{(t+1):\tilde{t}}) + \beta\}$
    Set $CP(\tilde{t}) = [CP(\bar{t}), \bar{t}]$
    Set $R_{\tilde{t}+1} = \left\{t \in R_{\tilde{t}} \cup \{\tilde{t}\} | F(t) + c(y_{(t+1):\tilde{t}}) + \beta \leq F(\tilde{t})\right\}$
end for
output : the vector of change-points $CP(n)$.

How to choose β

This problem can be viewed as a model selection problem.

A very common value for β is $\frac{p}{2}\log(n)$ also known as the BIC criterion, where p is the dimension of the parameters space.

References

Killick, R., Fearnhead, P., & Eckley, I. A. (2012, oct). Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, *107*(500), 1590–1598. doi: 10.1080/01621459.2012.737745

Truong, C. (2018). *Multiple change point detection – application to physiological signals.* (Theses, Université Paris Saclay). Retrieved from `https://tel.archives-ouvertes.fr/tel-01984997`