

Discussion on "Minimal penalties and the slope heuristic: a survey" by Sylvain Arlot

Titre: Discussion sur "Pénalités minimales et heuristique de pente" par Sylvain Arlot

Émilie Lebarbier¹

We study here the performance of the slope heuristics in a change-point detection framework.

Change-point setting. We observe a finite sequence $\{y_t\}_{t=1,\dots,n}$ realisation of independent variables Y_t drawn from a Gaussian distribution where the mean s is assumed to be piecewise constant on a partition m of $\llbracket 1, n \rrbracket$ (change-points delimiting the segments of the partition) and the variance be known:

$$Y_t \text{ ind. } \sim \mathcal{N}(s_r, \sigma^2) \quad \text{if } t \in \text{segment } r.$$

In such framework, the model selection issue arises for the choice of the number of segments (the size of m , $|m|$). To ensure that the selected estimator $\hat{s} = \hat{s}_{\hat{m}}$ satisfies an oracle inequality, Lebarbier (2005) shows that the optimal partition/segmentation \hat{m} minimises a penalized least-squares criterion $\hat{m} \in \underset{m}{\operatorname{argmin}} \|y - \hat{s}_m\|^2/n + \operatorname{pen}(m)$ where the penalty is

$$\operatorname{pen}(m) = \operatorname{pen}(|m|) = \sigma^2 \frac{|m|}{n} \left(c_1 \log \left(\frac{n}{|m|} \right) + c_2 \right) = \sigma^2 f_n(c_1, c_2, m)$$

In practice, since the penalty depends on the partition through its dimension, the optimal segmentation in K segments \hat{m}_K is computed for every K up to K_{\max} , then \hat{K} is obtained using the penalty.

Simulation design and quality criteria. We considered series of length $n = 200$ with $\sigma \in \{0.1, 0.5, 1, 1.5, 2\}$ (scenarios from easy to difficult detection). All series are affected by 4 change-points located at positions 60, 110, 140, 180. The mean within each segment alternates between 0 and 1, starting with $s_1 = 0$. Each combination was replicated $S = 200$ times. The quality of the segmentation parameter estimation is assessed via:

- * The difference between the true number of segments and the estimated one, $\hat{K} - K$;
- * The two components of the Hausdorff distance between the true and the estimated segmentations that are $d_1 = \mathcal{E}(m_s || m_{\hat{s}})$ and $d_2 = \mathcal{E}(m_{\hat{s}} || m_s)$ where $\mathcal{E}(m_A || m_B) = \sup_{b \in m_B} \inf_{a \in m_A} |a - b|$ in order to assess the quality of the change-point locations. d_2 assesses how our estimated segmentation is able to recover the true change-points. On the contrary, d_1 judges how relevant the

¹ UMR AgroParisTech/INRA MIA-Paris

proposed change-points are compared to the true segmentation.

* The risk of the proposed estimator $\hat{s}_{\hat{m}}$, $\|s - \hat{s}_{\hat{m}}\|^2$ in order to assess the global estimation of the mean s .

The criteria are also computed for the optimal segmentation for the true number of segments, denoted \hat{m}_K and we consider the optimal segmentation in terms of risk that is $\hat{m}_{\tilde{K}}$ where $\tilde{K} \in \operatorname{argmin}_K \|s - \hat{s}_{\hat{m}_K}\|^2$ (trajectorial oracle).

Expected results in change-point detection. A powerful segmentation procedure

* must recover the true segmentation in easy detection scheme leading to the selection of the true number of segments and with both null d_1 and d_2 ;

* tends to underestimate the number of segments in order to avoid false detection when the detection is difficult (selecting the true number of segments in this case is not desired). The obtained estimator yields then high values of d_2 due to the missed change-points but low values of d_1 as the change-points we propose tend to correspond to true ones. Note that on the contrary an overestimation of K results in a large d_1 and a small d_2 .

Algorithm 1 versus algorithm 2. Lebarbier (2005) proposed to take $c_1 = 2$ and $c_2 = 5$. Then the variance σ^2 can be seen as a constant α that can be calibrated using either algorithms 1 or 2.

Algorithm 1 (biggest jump). We study the impact of K_{\max} in this algorithm. Different values of $K_{\max} = n^\beta$ are considered with $\beta \in \{0.5, 0.6, 0.7, 0.8\}$. Results are given in Figure 1 and Table 1. Whatever K_{\max} , we observe the same tendency according to the detection difficulty: more the detection is difficult, less change-points are detected but they are well positioned. This behaviour is preferable (smaller d_1 compared to \hat{m}_K for $\sigma \geq 1$). The choice of K_{\max} has clearly an influence on the selection of the number of segments (thus on the segmentation): for a too small K_{\max} , the algorithm can detect no change-point whereas the detection is clear by eyes (see the example presented in Figure 4) and on the contrary for a too high value of K_{\max} , the number of segments is overestimated compared to smallest K_{\max} with thus too many spurious change-points (see the example presented in Figure 6). For $n = 200$ (our simulation scheme), it seems to be reasonable to choose $K_{\max} = n^{0.6}$. Indeed, in this case the algorithm 1 tends to recover the true number of segments, and therefore its performances are the same as that of \hat{m}_K , and the obtained estimator performs significantly better when the series become more difficult to segment. Moreover, even in the most difficult scenarios, our estimator has a performance close to that of the trajectorial oracle (orange in Figure 1).

Table 1 shows that the biggest jump can be reached for different values of α . According to results observed on simulations, the first one is preferred, i.e. the one associated to the smallest α (that is done here). We also observe that for $\beta = 0.6$, $\alpha = \sigma^2$ is well estimated except for the very easy detection case due to some simulations with no selected change-points. However when K_{\max} is too large, σ^2 is underestimated.

Algorithm 2 (slope). We propose to use this algorithm in two fashions: (i) estimate σ^2 by performing a linear regression of $\|y - \hat{s}_{\hat{m}_K}\|^2$ as a function of $f_n(c_1, c_2, m)$ and (ii) forgot the values of c_1 and c_2 and estimate them (including σ^2) by performing a regression of $\|y - \hat{s}_{\hat{m}_K}\|^2$ as a function of both $K \log(n/K)$ and K . We choose two sets of dimensions to run the regressions:

10 – 15 and 20 – 40. We have the same conclusions that for algorithm 1: (1) the underestimation of \hat{K} is desired in terms of segmentation when the detection problem is difficult whatever the strategy (compared to the true segmentation) and (2) the choice of the dimensions on which the regression is performed have an influence on the segmentation results (see also the examples presented in Figures 4, 5 and 6). The conclusion is not the same for the two strategies: for (i), it is preferable to choose smallest dimensions whereas this is the contrary for (ii) in terms of segmentation locations and in terms of risk.

Discussion. For both algorithms, calibrated choices need to be made. For easy detection cases (as in Figure 4 when eyes are sufficient), these choices are easy to make but they are quickly complicated with the detection difficulty as is common on real data. We can observe, for the algorithm 2 and the strategy (i) (see Examples presented in Figures 5 and 6) that there exists two schemes in the behaviour of $\|y - \hat{s}_{\hat{m}_K}\|^2$ as a function of $f_n(c_1, c_2, m)$: a first one after the oracle dimension (in red) followed by dimensions in which the 'noise is segmented' (in blue). This latter scheme is not preferable to perform the regression but as we can see the 'good' dimensions are difficult to identify.

To conclude I recommend the algorithm 1 for which the previous results militate for a calibration of K_{\max} and also of a K_{\min} .

Robustness to the model and comparison study. We compared the robustness of our criterion combined with the algorithm 1 (denoted Algo1) with two others: Lavielle (2005) simplified the penalty to βK and proposed a heuristic to calibrate β (denoted Lav) and Zhang and Siegmund (2007) developed a BIC criterion dedicated to the gaussian segmentation framework (denoted mBIC). We considered the same simulation design as previously but with a Student distribution for the noise with different degrees of freedom $\nu = \{50, 10, 6, 3\}$ ($\nu = 50$ being the closest Gaussian case). Results are given in Figure 3. When the simulation scheme is close to a Gaussian simulation, the performances of Algo1 and mBIC are slightly the same and better compared to Lav that tends to overestimate K : the underestimation of K results in a better performance in terms of segmentation (smaller d_1). When the degree of freedom of the Student distribution decreases, this tendency is accentuated for Algo1 and Lav leading to the same conclusion whereas the results of mBIC deteriorate. It is marked for the case $\nu = 3$ where Algo1 outperforms the others in terms of risk.

| β | 0.6 | | | | | 0.8 | | | | |
|------------------------|------|------|------|-----|-----|-------|------|------|-----|-----|
| σ | 0.1 | 0.5 | 1 | 1.5 | 2 | 0.1 | 0.5 | 1 | 1.5 | 2 |
| Jump size | 6.6 | 6.7 | 6.4 | 7 | 7.5 | 10 | 9.6 | 10 | 10 | 11 |
| Mult jump | 18% | 12% | 18% | 16% | 16% | 14% | 15% | 12% | 20% | 12% |
| $\sqrt{\alpha_{\min}}$ | 0.35 | 0.55 | 0.98 | 1.4 | 1.9 | 0.084 | 0.42 | 0.83 | 1.3 | 1.6 |

TABLE 1. Different values associated to the results presented in Figure 1 with $K_{\max} = n^{0.6}$ and $n^{0.8}$: Jump size: size in average of the biggest jump; Mult jump: percentage of simulations for which several jumps equals to the biggest jump dimension exist; α_{\min} is the value of α in average associated to the biggest jump.

References

- Lavielle, M. (2005). Using penalized contrasts for the change-point problem. *Signal Processing*, 85(8):1501–1510.
- Lebarbier, E. (2005). Detecting multiple change-points in the mean of Gaussian process by model selection. *Signal Processing*, 85:717–736.
- Zhang, N. R. and Siegmund, D. O. (2007). A modified Bayes information criterion with applications to the analysis of comparative genomic hybridization data. *Biometrics*, 63(1):22–32.

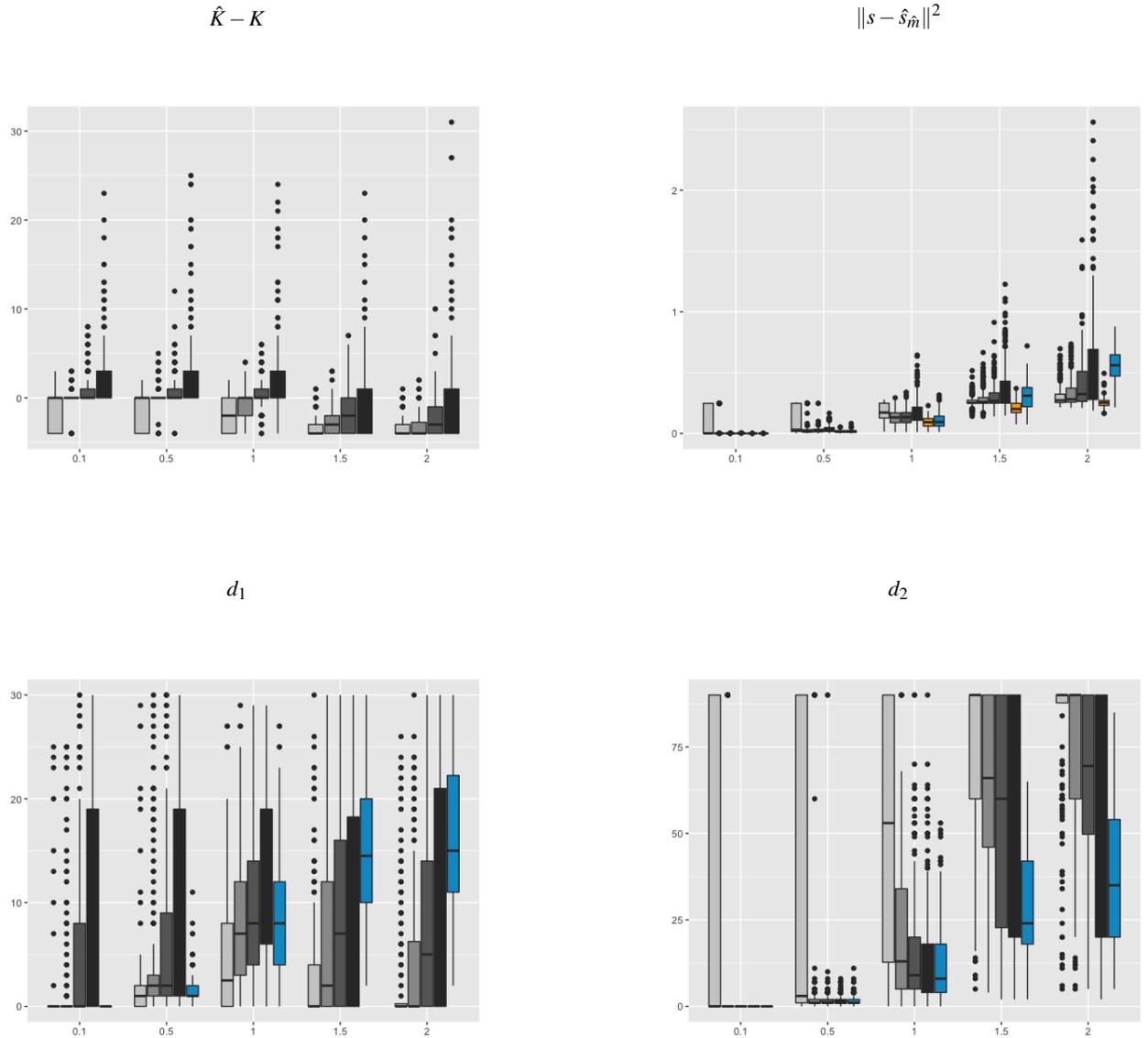


FIGURE 1. Boxplots of the different quality criteria. In each case, from left to right: the four left boxplots assess the estimator obtained with the algorithm 1 (grey) and the different values of $K_{max} = n^\beta$ with $\beta = 0.5, 0.6, 0.7, 0.8$ respectively, then the optimal segmentation $\hat{m}_{\hat{K}}$ (orange) and the optimal segmentation in K segments \hat{m}_K (blue). x-axis: σ .

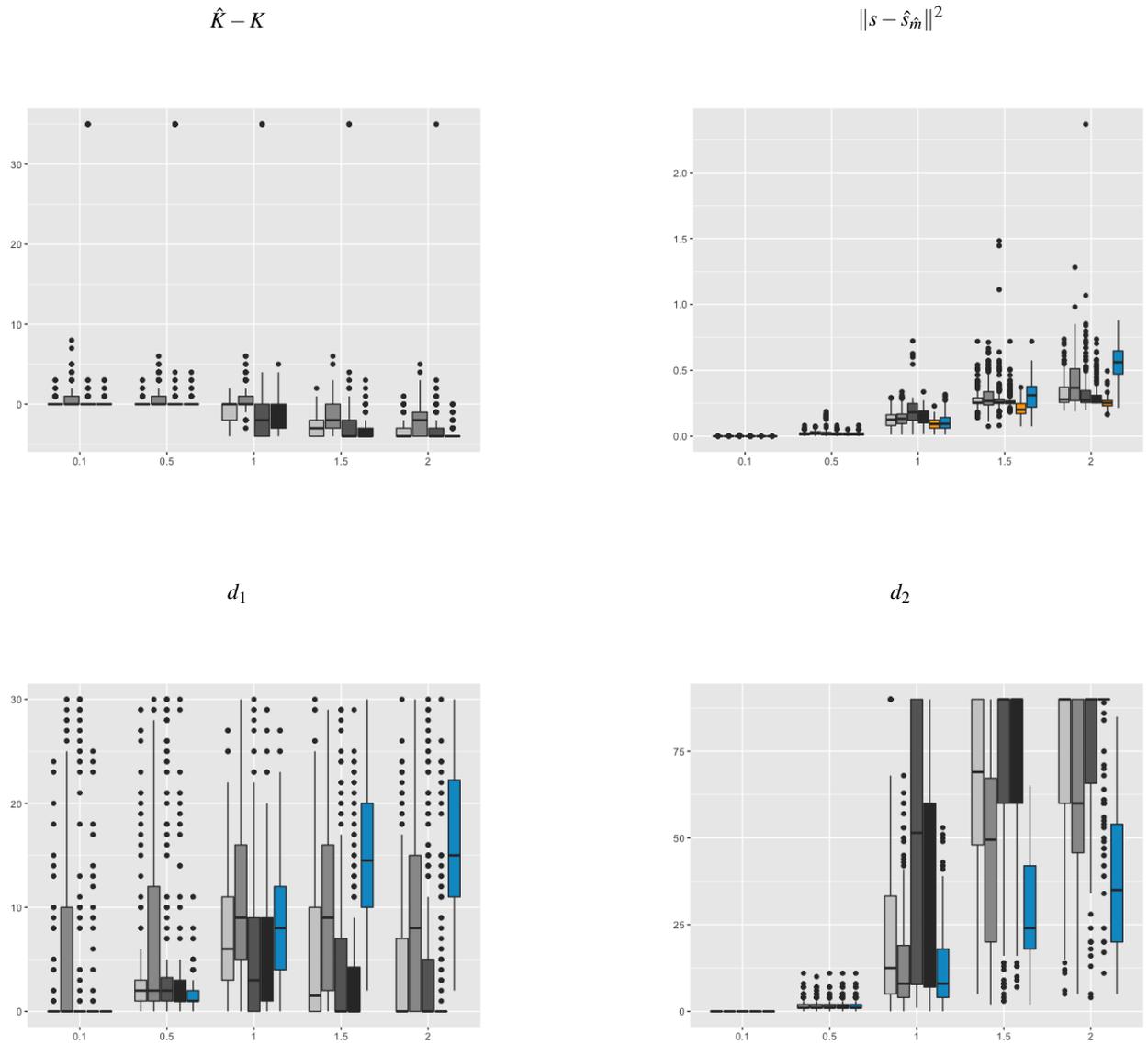


FIGURE 2. Same as in Figure 1 but with the algorithm 2. In grey from left to right: (i) 10-15; (i) 20-40; (ii) 10-15; (ii) 20-40.

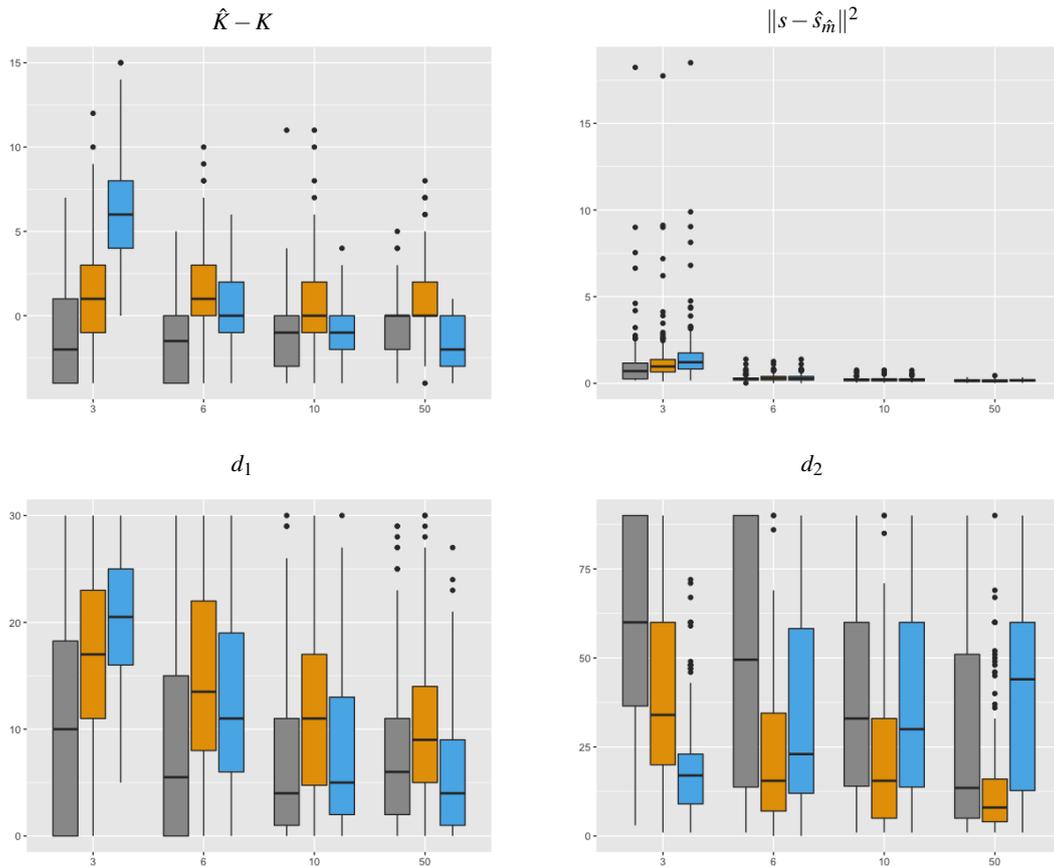


FIGURE 3. Comparison of our estimator obtained by algorithm 1 with $K_{max} = n^{0.6}$ (grey) with two others criteria Lav (orange) and mBIC (blue) for $n = 200$. x-axis: $\nu = 3$ (left) to 50 (right).

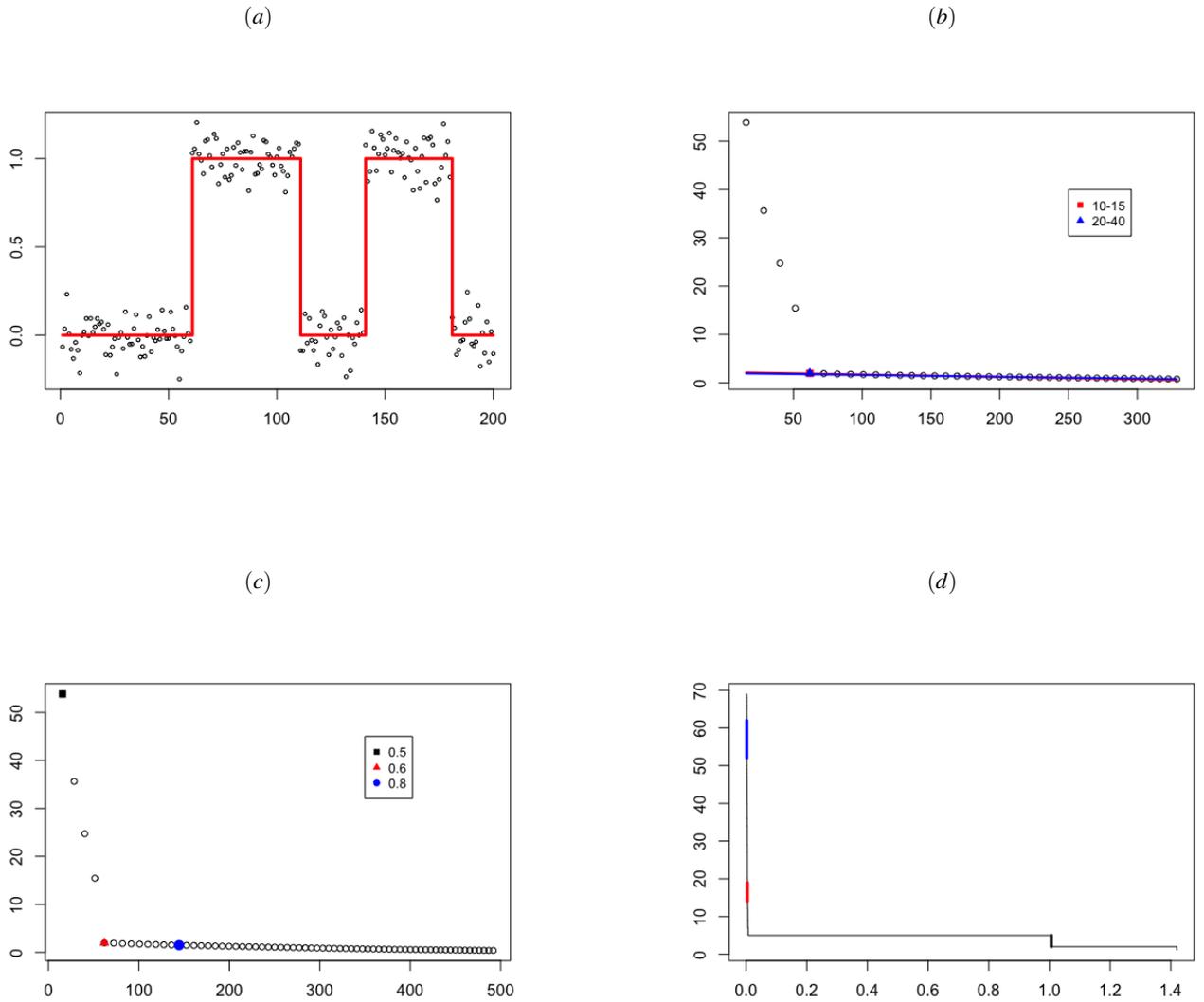


FIGURE 4. Example with $\sigma = 0.1$: (a) serie with the true mean (red); (b) plot of $\|y - \hat{s}_{\hat{m}_K}\|^2$ as a function of $f_n(c_1, c_2, m)$ with results of the algorithm 2 and the strategy (i): the selected number of segments is the same, $\hat{K} = 5$; (c) same as (b) with results of the algorithm 1 for $\beta = 0.4, 0.6, 0.8$: the selected number of segments is different, $\hat{K}^{\beta=0.5} = 1$, $\hat{K}^{\beta=0.6} = 5$ and $\hat{K}^{\beta=0.8} = 14$; (d) plot of $\alpha \mapsto \hat{K}_\alpha$ with the biggest jump associated to the results of (c).

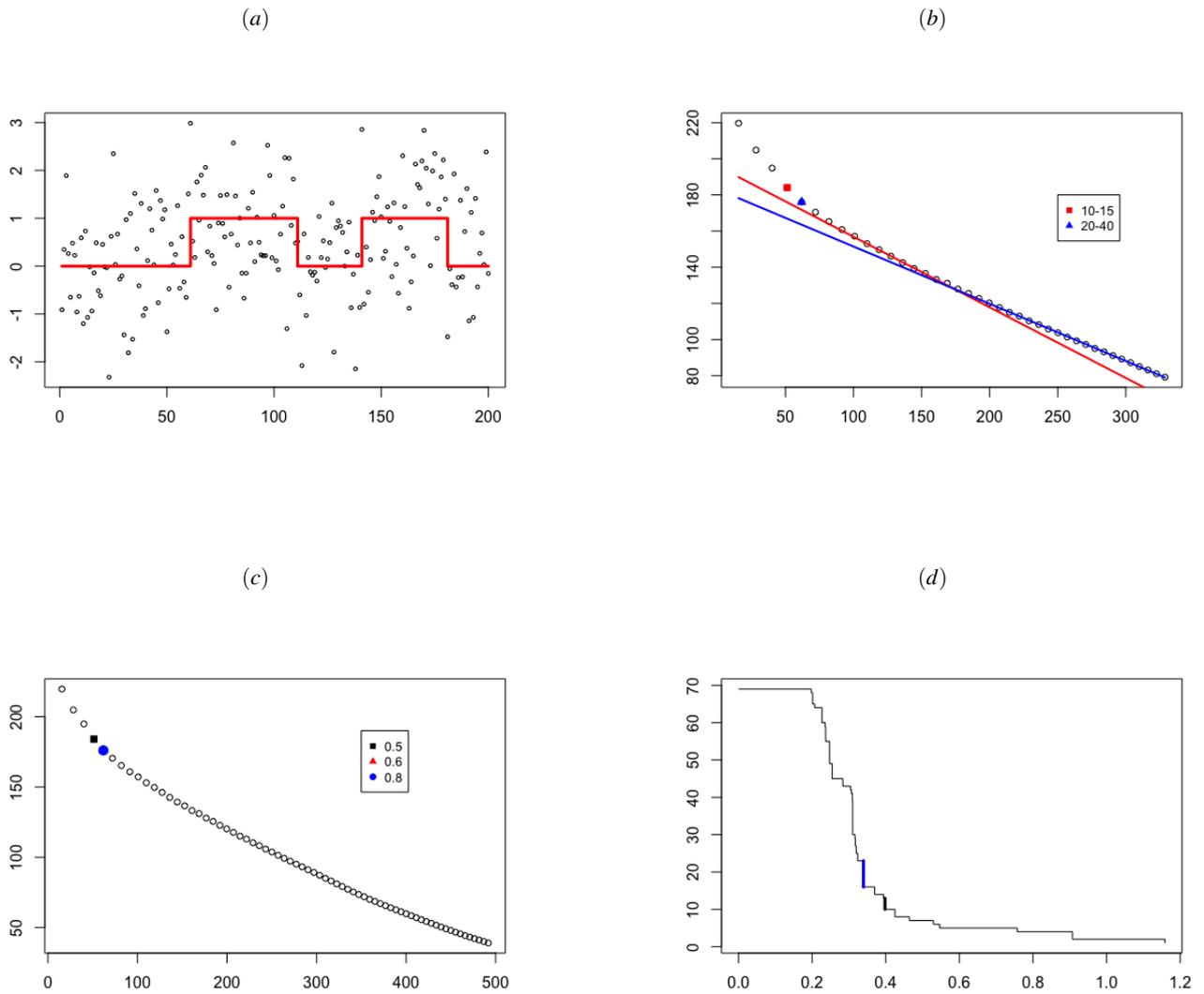


FIGURE 5. Example with $\sigma = 1$: (a) serie with the true mean (red); (b) plot of $\|y - \hat{s}_{\hat{m}_k}\|^2$ as a function of $f_n(c_1, c_2, m)$ with results of the algorithm 2 and the strategy (i): the selected number of segments are $\hat{K}^{10-15} = 4$ and $\hat{K}^{20-40} = 5$; (c) same as (b) with results of the algorithm 1 for $\beta = 0.5, 0.6, 0.8$: the selected number of segments is different, $\hat{K}^{\beta=0.5} = 4$ and $\hat{K}^{\beta=0.6} = \hat{K}^{\beta=0.8} = 5$; (d) plot of $\alpha \mapsto \hat{K}_\alpha$ with the biggest jump associated to the results of (c).

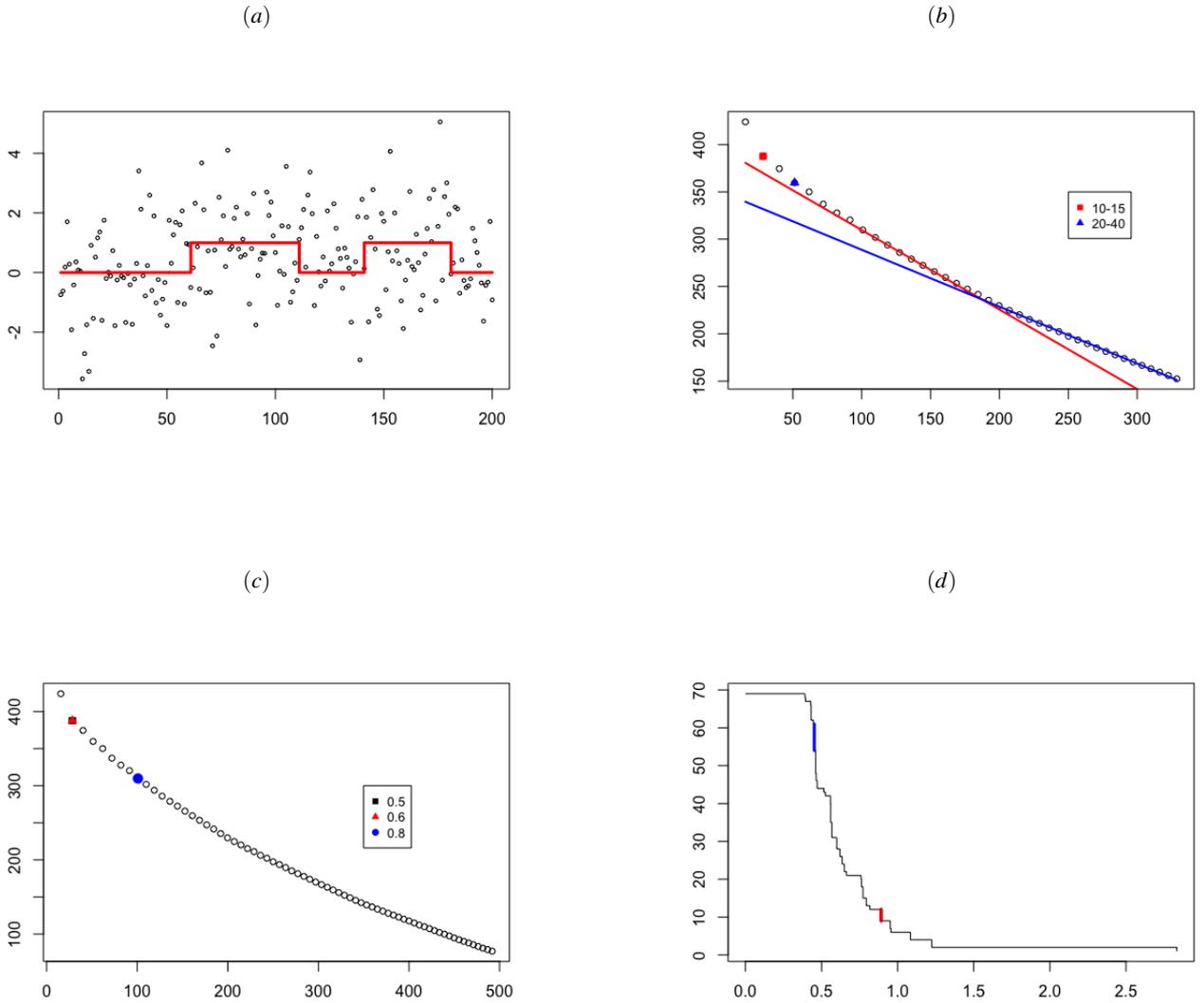


FIGURE 6. Example with $\sigma = 1.5$: (a) serie with the true mean (red); (b) plot of $\|y - \hat{m}_K\|^2$ as a function of $f_n(c_1, c_2, m)$ with results of the algorithm 2 and the strategy (i): the selected number of segments are $\hat{K}^{10-15} = 2$ and $\hat{K}^{20-40} = 4$; (c) same as (b) with results of the algorithm 1 for $\beta = 0.5, 0.6, 0.8$: the selected number of segments is different, $\hat{K}^{\beta=0.5} = \hat{K}^{\beta=0.6} = 2$ and $\hat{K}^{\beta=0.8} = 9$; (d) plot of $\alpha \mapsto \hat{K}_\alpha$ with the biggest jump associated to the results of (c).