Using CART to Detect Multiple Change Points in the Mean for large samples

by

Servane Gey and Emilie Lebarbier

Research Report No. 12
February 2008

Statistics for Systems Biology Group
Jouy-en-Josas/Paris/Evry, France
http://genome.jouy.inra.fr/ssb/

Using CART to Detect Multiple Change Points in the Mean for large samples

Servane Gey*, Emilie Lebarbier†

Abstract

A procedure is provided to detect multiple change-points in the mean of very large Gaussian signals. From an algorithmical point of view, visiting all possible configurations of change-points cannot be performed on large samples. The proposed procedure runs CART first in order to reduce the number of configurations of change-points by keeping the relevant ones, and then runs an exhaustive search on these change-points in order to obtain a convenient configuration. A simulation study compares the different algorithms in terms of theoretical performance and in terms of computational time.

Keyword : Change-points detection – Dynamic Programming – Model Selection – CART Algorithm.

1 Introduction

The aim of this paper is to propose a procedure which can be easily implemented to detect multiple change-points in the mean of a large Gaussian signal. However this procedure can be easily extended to density estimation by maximum likelihood. The main motivation comes from a practical point of view, as the detection of homogeneous areas in DNA sequence, which number of data can reach one million (see (Lebarbier 2002) for the application on DNA sequence). We take place in the now classical framework of penalized least-squares criterion, firstly developed by (Mallows 1974) and (Akaike 1973), (Akaike 1974). In this particular context, (Yao 1988), (Miao and Zhao 1993) estimate the number of change-points via the Schwarz's criterion (Schwarz 1978). More recently (Lavielle and Moulines 2000) propose to detect and estimate consistenly all those change points. In addition to this asymptotic point of view, (Lebarbier 2003)

*Laboratoire MAP5 - Université Paris V, 75270 Paris Cedex 06, France. Servane.Gey@math-info.univ-paris5.fr

†Département MMIP - AgroParisTech, 16 rue Claude Bernard, 75231 Paris Cedex 05, France. Emilie.Lebarbier@agroparistech.fr

adopts a nonasymptotic model selection approach based on the works of (Birgé and Massart 2001a). This procedure, called here exhaustive search, proposes to select the best configuration of change-points in term of quadratic risk among all possible configurations of change-points. In an algorithmical point of view, a dynamic programming is used to reduce the computational complexity of the exhaustive search from $\mathcal{O}(n^D)$ for a fixed number of change-points D to $\mathcal{O}(n^2)$ where n is the length of the observed signal. While this procedure performs well for moderatly sized signals, $n \leqslant 5000$ with our machine, it cannot be performed on much larger signals.

We propose a procedure which can extend the exhaustive search on large signals. Our aim is to reduce drastically the computational complexity without altering too much the accuracy of the estimation. Recall that the exhaustive search consider all possible configurations of change-points. Naturally some considered configurations are not relevant. Thus our idea is first to reduce the number of configuration of change-points by keeping the relevant ones and then to perform the exhaustive search on these change-points. In the literature, we can find fast procedures based on binary segmentation. The idea is to split at each step a segment into two segments by minimizing a criterion, and to stop when the criterions of the two obtained segments are lower than a threshold (see (Braun and Müller 1998) for example). The main difficulty of this kind of procedure is the choice of the threshold. A second drawback is that, unlike the exhaustive search, it only provides a local optimal segmentation : if a change-point is added or wrongly detected, i.e. if the procedure adds a false alarm, this false alarm has to be kept in the final configuration.

We propose a procedure combining CART (Classification And Regression Tree) developed by (Breiman et al. 1984), which is based on binary segmentation, and the exhaustive search. We call it the hybrid procedure. More precisely, it is performed in the two following steps :

1. First CART is applied. This gives some potential configurations of change-points instants that are revisited by the second stage of the procedure. It permits to reduce the collection of configurations of change-points by keeping the relevant ones. Let us note that we do not use the general methods proposed by Breiman *et. al*, which are based on test-sample or cross-validation (see (Breiman et al. 1984)). Indeed, since we work on a fixed regular grid, it would not be relevant to split the sample to obtain a test sample. Moreover, in term of computational time the cross-validation is considerably longer than the adaptive method proposed by (Birgé and Massart 2001b) that we use in this paper. Let us note that, unlike the above mentioned binary segmentation methods, CART avoids the problem of the choice of a threshold. This procedure is computationally faster than the exhaustive search (often of order $\mathcal{O}(n\log(n))$). Nevertheless, as mentioned above about sequential algorithm, it may add some false alarms.

2. The role of the second step is to remove the false alarms which can be added by the CART procedure by performing the exhaustive search on the configuration of change-points provided by CART.

The paper is organized as follows. Section 2 describes the model and the notations and recall basics about penalized least-squares model selection from Birgé and Massart's context. Section 3 deals with the exhaustive search and CART procedures; its gives for each the considered collection of partitions and the penalized criterion. The hybrid procedure is presented in Section 4 while a simulation study is performed in Section 5 to involve its expected behaviour. Sections 6 and 7 are appendices, on the one hand, highlighting the method used to calibrate the penalty constant for each penalized criterion and, on the other hand, giving the proofs of the computational complexities of the exhaustive search and CART procedures.

2 Preliminaries and Notations

Let us consider the following change-points detection problem : the observed sequence $(y_1, \ldots, y_n)$ is supposed to be of the form

$$y_t = f(t) + \varepsilon_t, \quad \text{for } t = 1, \ldots, n, \tag{2.1}$$

where f is assumed to be piecewise constant :

$$f = \sum_{k=1}^{K} s_k \mathbb{1}_{]\tau_{k-1}, \tau_k]}, \tag{2.2}$$

with for each k, $s_k \in \mathbb{R}$, and $\mathbb{1}_I(x) = 1$ if $x \in I$ and 0 otherwise. Errors (ε_t) are supposed to be zero-mean, identically distributed unobservable Gaussian independent random variables of common variance σ^2. In a change-points detection issue, the (τ_k) represent the so-called change-point instants, $K-1$ their number and the (s_k) the means between the change-points. These parameters are supposed to be unknown and our goal is to estimate them via the estimation of f from $(y_1, \ldots, y_n)$ by

$$\hat{f} = \sum_{k=1}^{\hat{K}} \hat{s}_k \mathbb{1}_{]\hat{\tau}_{k-1}, \hat{\tau}_k]}.$$

The estimated change-points correspond to the points of discontinuity $(\hat{\tau}_k)$ of $\hat{f}$. Hence we will assimilate the estimator of the function f and the estimator of the configuration of change-points.

The estimation method we adopt here takes place in the now classical context of penalized least-squares minimization. For a sake of completeness, let us introduce some notations used in the sequel and recall shortly the basics of the method.

Let us consider some collection $\mathcal{M}_n$ of partitions of $\{1,\dots,n\}$, each partition m in $\mathcal{M}_n$ having D_m pieces corresponding to a configuration of D_m-1 change-points. Then for each m the minimum least-squares estimator of f is

$$\hat{f}_m = \underset{\{u\ ;\ u=\sum_{I\in m} u_I \mathbb{1}_I\}}{\operatorname{argmin}} \frac{1}{n}\sum_{t=1}^{n}(y_t-u(t))^2 = \sum_{I\in m}\overline{y}_I \mathbb{1}_I,$$

where $\overline{y}_I$ is the empirical mean of y on I. Then we have at hand a collection of estimators $(\hat{f}_m)_{m\in\mathcal{M}_n}$ and the goal is to select a final estimator $\tilde{f}$ among this collection. This is done by choosing some partition $\hat{m}$ and setting $\tilde{f}=\hat{f}_{\hat{m}}$, where $\hat{m}$ minimizes the penalized least-squares criterion

$$crit_n(m) = \frac{1}{n}\sum_{I\in m}\sum_{t\in I}(y_t-\overline{y}_I)^2 + pen_n(D_m), \tag{2.3}$$

where pen_n is a penalty function, positive and nondecreasing of D_m.

We recall the result of (Birgé and Massart 2001a) in the Gaussian case in order to highlight the penalties used in the two methods we consider in the next section : if pen_n verifies for each $m\in\mathcal{M}_n$

$$pen_n(D_m) \geqslant K\sigma^2\frac{D_m}{n}\left(1+\sqrt{2L_m}\right)^2, \tag{2.4}$$

with $K>1$ and $(L_m)_{m\in\mathcal{M}_n}$ such that

$$\sum_{\{m\in\mathcal{M}_n\ ;\ D_m>0\}} e^{-L_m D_m} < +\infty, \tag{2.5}$$

then the quadratic risk of $\tilde{f}$, $\mathbb{E}\left[\|f-\tilde{f}\|^2\right]$, is close to $\inf_{m\in\mathcal{M}_n}\mathbb{E}\left[\|f-\hat{f}_m\|^2\right]$, where $\|.\|$ is the $\mathbb{L}^2([1,n])$-norm. The weights $(L_m)_{m\in\mathcal{M}_n}$ in the penalty are generally chosen in such a way that they only depend on the dimension D_m. Then the condition (2.5) can also be written as

$$\sum_{D\geq 1}\#\{m\ ;\ D_m=D\}e^{-L_D D} < +\infty.$$

From this point of view, pen_n heavily depends on n and on the size of the subsets $\{m\in\mathcal{M}_n\ ;\ D_m=D\}$, $D\geq 1$. Moreover, since the penalty depends on the partition only via its dimension, the selection of $\hat{m}$ can be done in 2 steps. For a fixed dimension D, compute

$$\hat{m}_D = \underset{\{m\in\mathcal{M}_n\ ;\ D_m=D\}}{\operatorname{argmin}} \frac{1}{n}\sum_{I\in m}\sum_{t\in I}(y_t-\overline{y}_I)^2, \tag{2.6}$$

that leads to the subfamily $\widetilde{\mathcal{M}}_n=\{\hat{m}_D\ ;\ 1\leqslant D\leqslant n\}$ of $\mathcal{M}_n$, containing at most one partition per dimension. Then $\tilde{f}$ is obtained as $\tilde{f}=\hat{f}_{\hat{m}_{\hat{D}}}$, where

$$\hat{D} = \underset{\{D\ ;\ \hat{m}_D\in\widetilde{\mathcal{M}}_n\}}{\operatorname{argmin}}\left[\frac{1}{n}\sum_{I\in\hat{m}_D}\sum_{t\in I}(y_t-\overline{y}_I)^2 + pen_n(D)\right].$$

3 Two Procedures

We propose to use two procedures to estimate the function f. The first one is an exhaustive search based on the natural idea to consider all possible partitions of $\{1,\dots,n\}$ (see (Lebarbier 2003) for the Gaussian case). The second one is CART, proposed by (Breiman et al. 1984), that considers some relevant subcollection of partitions. We will see that $\widetilde{\mathcal{M}}_n$ depends on the procedure, so we denote this quantity by $\widetilde{\mathcal{M}}_n^{(\mathrm{es})}$ and $\widetilde{\mathcal{M}}_n^{(\mathrm{cart})}$ for the exhaustive search and for CART respectively. We give here a short summary of these two procedures.

3.1 Exhaustive Search

Let $\mathcal{M}_n^{(\mathrm{es})}$ be the family of all possible partitions of the grid $\{1,\dots,n\}$ where n is the size of the sample. Remark that $\mathcal{M}_n^{(\mathrm{es})}$ is the maximal family of partitions of $\{1,\dots,n\}$. For a given dimension D, the best D-dimensional partition among $\mathcal{M}_n^{(\mathrm{es})}$ is constructed using dynamic programming to reduce the computational load from $\mathcal{O}(n^D)$ to $\mathcal{O}(n^2)$, leading to the expected family $\widetilde{\mathcal{M}}_n^{(\mathrm{es})}$ (for example see (Kay 1998) for an overview of dynamic programming). The next step is to determine a penalty function that leads to a final estimator $\tilde{f}$ convenient in terms of quadratic risk. (Lebarbier 2003) shows that the penalty function

$$pen_n(D_m) = \frac{D_m}{n}\sigma^2\left(2\log\frac{n}{D_m}+5\right) \tag{3.7}$$

performs well in a theoretical point of view. The two fine tuned constants 2 and 5 are obtained via numerical simulations. The $\log(n/D_m)$ term comes from the fact that

$$\#\{m\ ;\ D_m = D\} = \binom{n-1}{D-1},$$

so taking $L_D = \log(n/D)$ satisfies (2.5), and leads to this form of penalty by (2.4). In practice the noise variance σ^2 is unknown and an alternative method is recalled in Section 6 to avoid its estimation.

3.2 CART for Regression

The CART procedure (Breiman et al. 1984) is primarily computed in two steps.

- The first step, called the growing procedure, is sequential. It consists in splitting at each step the considered segment into two segments by minimizing the sum of the least-squares criterions of the segments. The split stops when less than l_{min} points are left in each resulting segment. This collection is called *maximal tree* and is usually represented by a binary tree, where each node corresponds to a split. For example, see the bottom of Figure 1 : the first

change-point is detected at time 17 and corresponds to the root of the tree; then the two following change-points are detected at times 15 and 30 and correspond to the left and right following nodes of the tree respectively. And so on until all the points are visited (here $l_{min} = 1$).

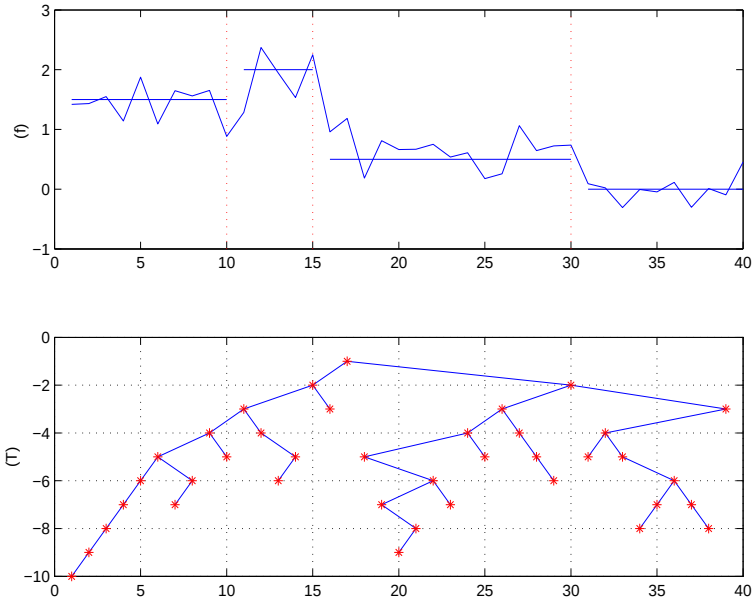

Figure 1: Example of a signal (f) and its associated CART tree (T).

Then the collection of partitions obtained from the maximal tree, $\mathcal{M}_n^{(\text{cart})}$, is included in $\mathcal{M}_n^{(\text{es})}$ and a partition of $\mathcal{M}_n^{(\text{cart})}$ corresponds to a *pruned subtree*, that is any binary subtree of the maximal one containing its root. In the example, the partition defined by the points at times 17, 15 and 30 is a subtree pruned from the maximal one, but the partition defined by the points at times 15 and 30 is not, since the root at time 17 must belong to the subtree.

• The second step, called the pruning procedure, avoids the computation of all D-dimensional pruned subtrees by selecting directly the relevant partitions. This procedure is closely related to the penalty function

$$pen_n(m) = \beta \frac{D_m}{n},$$

where β is an unknown constant. This penalty comes from the fact that the number of pruned subtrees of dimension D is of the order of $(en/2)^2 D$ (see (Gey and Nedelec 2001) for more details). The general strategy of the pruning procedure is then to make β increase in the corresponding penalized criterion (2.3) so that D_m decreases (see (Breiman et al. 1984) for more details). This leads to a collection of nested trees $(m_i)_{1 \leqslant i \leqslant K_T}$, with $m_{K_T} = [1, n]$, associated with an increasing sequence of $(\beta_i)_{1 \leqslant i \leqslant K_T}$, with $\beta_1 = 0$.

In order to choose a tree among the collection $(m_i)_{1 \leqslant i \leqslant K_T}$, i.e. to reach a suitable value of β, we use the method given in Section 6.

4 The Hybrid Procedure

4.1 Motivation

We focus here on the advantages and drawbacks of the two above mentioned procedures to motivate the hybrid procedure. We first give the computational complexities of each procedure in order to cast some light on the interest of our approach.

Proposition 4.1. *Assume that we have at hand a sample of size n. Then the complexity $C(n)$ of the exhaustive search procedure by dynamic programming is*

$$C(n) = \mathcal{O}(n^2). \tag{4.8}$$

Proposition 4.2. *Assume that we have at hand a sample of size n. Let us denote $n_t \leqslant n$ the number of nodes of the deepest tree constructed during the first step of the CART procedure. Let $C_1(n, n_t)$ and $C_2(n_t)$ be the respective complexities of the growing and pruning procedures. Then we have*

$$\mathcal{O}(n \log_2 n_t) \leqslant C_1(n, n_t) \leqslant \mathcal{O}(n n_t), \tag{4.9}$$

$$\mathcal{O}(n_t) \leqslant C_2(n_t) \leqslant \mathcal{O}(n_t^2). \tag{4.10}$$

The proofs of Propositions 4.1 and 4.2 are given in Section 7.

Remark 1. Let us notice that C_2 only depends on the number n_t of nodes of the deepest tree, whereas C_1 depends on n_t and n, what is expected since the pruning procedure is performed on the maximal fixed tree. Actually, what is really important is the sum of these two computational complexities. Furthermore, let us remark that the largest computational complexity for the two combined procedures used to construct a suitable tree is the same as the one of the exhaustive search. To reach this computational complexity with CART, it is necessary to obtain a complete thread-like binary tree having exactly n nodes after the growing procedure. In this case, it is clear that we do not improve the computation time and we lose in accuracy. However, during the simulations we have done in practice, this case has never been observed (see Section 5 for numerical results).

From the procedures and their computational complexities given above, we can make the following observations.
On one hand, it is shown in (Lebarbier 2003) that the exhaustive search leads to an estimator of f optimal in term of risk. Nevertheless, as shown by Proposition

4.1, its computational complexity of order $\mathcal{O}(n^2)$ does not allow to perform this procedure on too large samples. Typically, in our framework, the size of the sample cannot be larger than $n = 5000$.
On the other hand, Proposition 4.2 suggests to use CART on large samples since its computational complexity is of order $\mathcal{O}(n \log n)$. Nevertheless, CART may add some nonexistant change-points, or false alarms, in the estimated change-points, as shown on Figure 1 for the first change-point at time 17. Indeed, to catch the true change-points at times 10, 15 and 30 detected further in the tree, CART will be forced to keep the first one in the final estimator.

So the principle of the hybrid procedure is to first perform CART on the signal to fastly obtain a relevant configuration of change-points, and then to run the exhaustive search in order to remove the false alarms.

4.2 Description

First let us give a precise description of the hybrid procedure computed in three steps :

1. CART is performed and a partition of dimension $\hat{D}_c$ is obtained by the heuristic given in Section 6.
2. We consider the $v\hat{D}_c$-dimensional subtree in the sequence $(m_i)_{1 \leqslant i \leqslant K_T}$, where v is an integer greater than one. The corresponding change-points are then identified to a new family $\mathcal{L}$ of potential change-points.
3. We have at hand the collection of partitions $\mathcal{M}_{n,\mathcal{L}} = \mathcal{P}(\mathcal{L})$ obtained from the new grid $\mathcal{L}$. The exhaustive search is performed on this collection and provides the final estimator $\tilde{f}$.

Remark 2. We take the subtree having $v\hat{D}_c$ leaves in order to catch relevant change-points instants that could be eventually missed or shifted by the first selection. Let us notice that if the ratio $\hat{D}_c/n$ is small, according to the expected value of $\hat{D}_c$, one can choose for example $v = 4$. On the other hand, it is clear that if the ratio is close to one or if the value $\hat{D}_c$ is larger than the expected value, then setting $v = 1$ is natural. Let us remark that this value should not be chosen too large to keep the interest of performing CART in a first step.

We propose an illustration of the different steps of this procedure and then compare its performance with the two other ones.

4.3 Illustration

A sequence of $y = (y_1, \ldots, y_n)$ is simulated from (2.1) with $n = 1000$ according to a function f_1 plotted in Figure 2, and a noise variance $\sigma^2 = 1$. The observed

serie is plotted in Figure 3.

Remark 3. We take for this simulation $n = 1000$, which allows us to perform the exhaustive search in order to compare its performances with the ones of the hybrid procedure. However, the interest of the hybrid procedure is obviously to be run on larger samples.

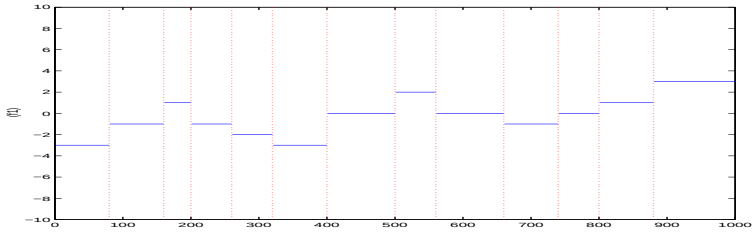

Figure 2: Function f_1.

We apply the three proposed procedures on this realization. The penalized estimators are plotted in Figures 3 and 4. To show the dynamic of the hybrid procedure, we give a short description of the different results : first, the penalized estimator obtained by CART is plotted in Figure 3-(a). The dimension of its associated partition is $\hat{D}_c = 15$. We take $v = 4$ and the corresponding subtree is displayed in Figure 3-(c). Then an exhaustive search is performed on the new associated grid and its penalized estimator is plotted in Figure 3-(b). The symbols represented on the tree in Figure 3-(c) (o, $*$ and $+$) correspond respectively to the change-points removed, kept and added after running the exhaustive search on the tree given by CART.

First of all, we can notice on this example that the hybrid procedure behaves as expected, i.e. that the exhaustive search allows to remove the change-points added by CART and to catch the missed or shifted ones : for example, CART detects two change-points at 209 and 404 and then keeps the 15-dimensional tree to reach the true change-points at 200 and 400. Then the exhaustive search removes the change-points at 209 and 404 and keeps the two other ones. Moreover four change-points at times 75, 564, 636 and 876 selected by CART are then shifted to 79, 560, 663 and 880, which are closer to the true ones. Furthermore, let us remark that the hybrid procedure selects exactly the same change-points as the exhaustive search, except one in the neighborhood of 660.

On the other hand, if we compute the loss $\|f_1 - \tilde{f}_1\|^2$ of each penalized estimator $\tilde{f}_1$ provided by the three procedures, we find 0.11, 0.04 and 0.038 respectively for CART, hybrid and exhaustive search. So the hybrid procedure improves the performance of CART and does not really alter the ones of the exhaustive search on this example. Let us notice that the penalized estimators obtained by the hybrid and the exhaustive search are the penalized estimators of minimal loss among the corresponding and respective collections of partitions.

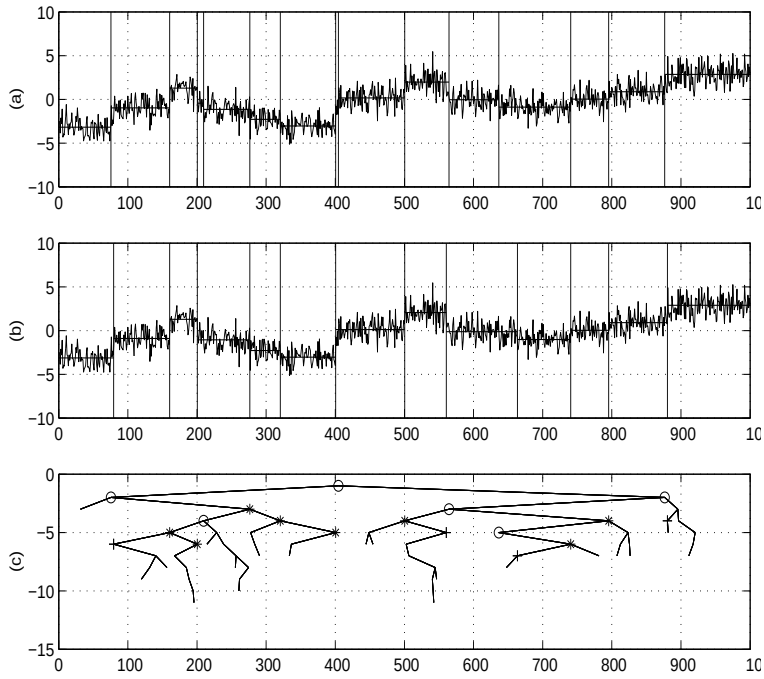

Figure 3: Penalized estimators of f_1 obtained respectively by (a) CART and (b) hybrid procedures. On (c), the tree on which the exhaustive search is performed in the hyvrid procedure, where o : removed change-points, $*$: kept change-points and $+$: added change-points

5 Simulation Studies

The purpose of this section is to compare the performance of the three considered procedures. This performance is evaluated by the risk function of each penalized estimator and the computational time needed by each procedure. Let us denote by

- $R_{(.)}$ the quadratic risk of the estimator $\tilde{f}$ provided by the procedure (.) : $R_{(.)} = \mathbb{E}\left[\|f - \tilde{f}_{(.)}\|^2\right]$. Since exact value of this risk can not be reached analytically, it is estimated via a Monte Carlo method, averaging the values of $\|f - \tilde{f}_{(.)}\|^2$ over N samples. Furthermore, since $R_{(es)}$ is the minimum of the risks in the cases where the true function is piecewise constant, we take it as a reference for the other risks.
- $cput_{(.)}$ the average computation time of the (.) procedure, given in seconds. In a same way, it is estimated by $\frac{1}{N}\sum_{j=1}^{N} cput_{(.)}^{(j)}$ where $cput_{(.)}^{(j)}$ is the computation time of the procedure (.) for the jth sample.

This simulation study is performed with Matlab6 scientific software on an Ultra 10-440 MHz SUN workstation. We take the following parameters : $n = 1000$,

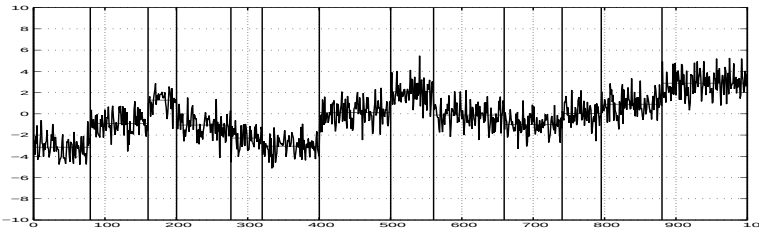

Figure 4: Penalized estimator of f_1 obtained by exhaustive search.

$\sigma^2 = 1$ and 3 functions f_1, f_2 and f_3 respectively plotted in Figures 2 and 5. Let us remark that the function f_3 is not piecewise constant.

Concerning the free parameters, $l_{min} = 1$ in CART to ensure that close change-points will be detected, and the regression used in the heuristic method for CART (see Section 6) is made on dimensions from 20 to 40. Furthermore, we set $v = 4$ in the hybrid procedure. Moreover, the number of simulated samples to estimate the considered values is $N = 300$.
The results are given in Table 1.

	f_1	f_2	f_3
$R_{(cart)}/R_{(es)}$	1.28	1.198	0.986
$R_{(hyb)}/R_{(es)}$	1.085	1.007	1.017
$cput_{(cart)}$	2.42	2.5	2.44
$cput_{(hyb)}$	3.01	2.83	2.79
$cput_{(es)}$	26.5	25.95	26.14

Table 1: Estimations of the penalized estimators risk and of the computational time of each procedure for the three proposed functions.

One observes the following :

1. The estimators obtained by the hybrid procedure are close in terms of risk from the estimators obtained by an exhaustive search on the whole sample,

2. the hybrid procedure takes much less operations in an procedureical point of view than an exhaustive search,

3. let us notice that the CART estimator of f_3 has a smaller risk than the one of the exhaustive search. This phenomenon can be explained by the fact that CART acts locally while the others search in a more global way; so CART will add some change points to better approximate the polynomial part of the function, while the others will lose in approximation to gain in variance in a more global way. However, the risk of the estimators are closer.

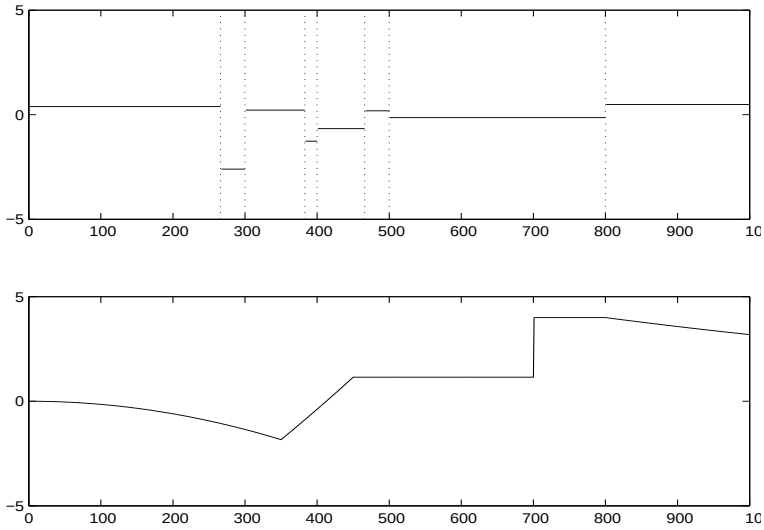

Figure 5: Functions f_2 and f_3

Acknowledgements

We would like to thank Marc Lavielle for his help on programming issues. We are also grateful to Jean-Michel Poggi and Pascal Massart for helpful discussions.

6 Appendix 1

6.1 Calibration of the Penalty

When the penalty has the general form

$$pen_{\alpha,n}(D) = \alpha f_n(D), \qquad D \geqslant 1,$$

where $f_n(D)$ is well defined ($f_n(D) = D/n\,(2\log(n/D) + 5)$ for the exhaustive search and $f_n(D) = D/n$ for CART), the problem is to find a suitable value for α.
We use a heuristic method, based on the work (Birgé and Massart 2001b), that is recalled here. The idea of the heuristic is that when the considered partition is high-dimensional, its approximation error is close to zero, so the corresponding criterion will represent an estimation of the penalty. The basic principle is to fit a linear regression of $\gamma_n(\hat{f}_{\hat{m}_D}) = (1/n)\sum_{I\in\hat{m}_D}\sum_{t\in I}(y_t - \overline{y}_I)^2$ with respect to $f_n(D)$ for large D and use the estimated regression coefficient $-\hat{\alpha}$ as an estimator of α. Then, in order to get a convenient penalty, it suffices to take $pen_{2\hat{\alpha},n}$.

However, it may be difficult to choose the dimensions between which the linear

regression of $\gamma_n(\hat{f}_{\hat{m}_D})$ with respect to $f_n(D)$ is done, so this heuristic cannot be performed so easily. Through theoretical results and some practical observations (see (Birgé and Massart 2001b)), the following heuristic method can be performed : for a fixed α, consider the partition of dimension $\hat{D}_\alpha$ defined by

$$\hat{D}_\alpha = \text{argmin}_{D \geqslant 1} \left\{ \gamma_n(\hat{f}_{\hat{m}_D}) + pen_{\alpha,n}(D) \right\}.$$

Then the idea is to increase slowly α from 0 and compute the corresponding models of dimensions $(\hat{D}_\alpha)_{\alpha \geqslant 0}$. One can observe a big jump in the dimensions when α reaches a threshold $\hat{\alpha}$. The penalty function will then be taken as $pen_{2\hat{\alpha},n}$.

6.2 Application to each Procedure

Application to the exhaustive search

According to the previous paragraph, we compute the function $\alpha \to \hat{D}_\alpha$ (plotted in the left side of Figure 6) and choose $\hat{\alpha}$ as the one associated with the big jump of dimensions. The user should choose the maximal dimension, i.e a minimal value of α to perform this method (see (Lebarbier 2003)).

Application to CART

Up to now, general methods used in CART to choose a tree among the sequence $(m_i)_{1 \leqslant i \leqslant K_T}$ are based on test-sample or cross-validation (see (Breiman et al. 1984)). However, in our framework, since we are working on a fixed grid, it would not be relevant to use a method splitting the sample as the one based on test-sample. Moreover, in term of computational time, the cross-validation based method is considerably longer than the heuristic proposed above.
In practice, for large n, we observe that, beyond some dimension, $n\gamma_n(\hat{f}_{m_i})$ becomes an affine function of the number of segments (see Figure 6). The choice of the dimensions between which we fit the regression is not really important as long as they are after the relevant point where $n\gamma_n(\hat{f}_{m_i})$ becomes linear. That is why this choice is let to the user.

7 Appendix 2

COMPLEXITY OF THE EXHAUSTIVE SEARCH PROCEDURE 4.8:

The complexity of the procedure is the sum of the three complexities :

1. The collection of estimators $\{\hat{f}_{\hat{m}_D}, D = 1, \ldots, n\}$ is obtained from a dynamic programming which has a complexity of $\mathcal{O}(n^2)$.

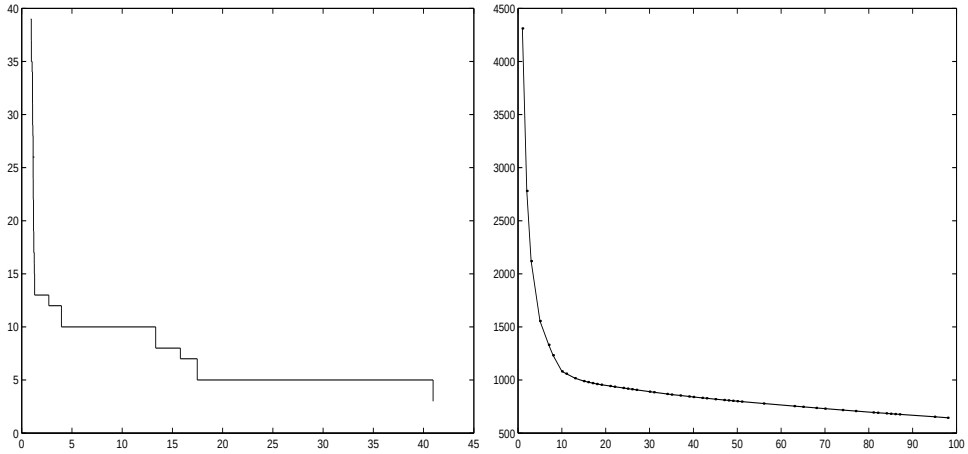

Figure 6: left : Function $\alpha \rightarrow \hat{D}_\alpha$, $\alpha \geqslant 0$ (exhaustive search). Right : Function $D_i \rightarrow n\gamma_n(\hat{f}_{m_i})$, $i = 1, \ldots, K_T$ (CART).

2. The computational complexity of the function $\alpha \rightarrow \hat{D}_\alpha$ is $\mathcal{O}(n)$. Morevover the estimation of α only needs one operation. Then its complexity is of the order $\mathcal{O}(1)$.

3. Since the best partition is selected among the collection $\{\hat{m}_D, D = 1, \ldots, n\}$, the complexity of this step is $\mathcal{O}(n)$.

The complexity of the exhaustive search is then $\mathcal{O}(n^2)$.

COMPLEXITY OF THE GROWING PROCEDURE 4.9:

Since the complexity of this procedure depends on the form of the constructed tree, it can not be written in a close form. However, it can be bounded by considering two cases :

- In the best situation, the constructed tree is completely balanced. Let us denote by h the depth of the deepest tree. Then $n_t = 1+2+3+\ldots+2^{h-1} = 2^h - 1$. Since this procedure is recursive we have the following relationship

$$C_1(n, n_t) = n + 2C_1(\left[\frac{n}{2}\right], \left[\frac{n_t}{2}\right])$$

  with $C_1(j, 1) = j$. We then obtain easily that $C_1(n, n_t) = nh = n \log_2 n_t - 1 = \mathcal{O}(n \log_2 n_t)$.

- In the worst situation, the tree is a thread-like one. In other words, the tree has one node at each depth. So, as above, we have the relationship

$$C_1(n, n_t) = n + C_1(n-1, n_t - 1)$$

  with $C_1(j, 1) = j$. So $C_1(n, n_t) = \sum_{i=1}^{n_t+1}(n-i) = \mathcal{O}(nn_t)$.

COMPLEXITY OF THE PRUNING PROCEDURE 4.10:

The pruning procedure depends only of the number of nodes in the deepest tree. We have therefore the two extreme cases :

- In the best case, the first subtree pruned from the deepest one is the root. So it is easy to see that in this case the complexity is reduced to the number of nodes, i.e to $\mathcal{O}(n_t)$.
- In the worst case, the pruning procedure goes leaf by leaf from the deepest tree to the root, so the number of subtrees contained in the resulting sequence is n_t. Then, we have the following relationship

$$C_2(n_t) = n_t + C_2(n_t - 1)$$

with $C_2(1) = 1$.
So $C_2(n_t) = \sum_{i=1}^{n_t} i = \mathcal{O}(n_t^2)$.

References

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In *Second International Symposium on Information Theory (Tsahkadsor, 1971)*, pp. 267–281. Budapest: Akadémiai Kiadó.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Trans. Automatic Control AC-19*, 716–723. System identification and time-series analysis.

Birgé, L. and P. Massart (2001a). Gaussian model selection. *J. Eur. Math. Soc. 3*, 203–268.

Birgé, L. and P. Massart (2001b). A generalized C_p criterion for Gaussian model selection. Technical report, Publication Université Paris-VI.

Braun, J. and H. Müller (1998). Statistical methods for dna sequence segmentation. *Statistical Science 13*(2), 142–162.

Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone (1984). *Classification And Regression Trees*. Chapman & Hall.

Gey, S. and E. Nedelec (2001). Model selection for CART regression trees. Technical Report 56, Université Paris XI. To appear in IEEE Transaction of Information Theory.

Kay, S. M. (1998). *Fundamentals of statistical signal processing - Detection theory*, Volume II. Prentice Hall signal processing series.

Lavielle, M. and E. Moulines (2000). Least Squares estimation of an unknown number of shifts in a time series. *Jour. of Time Series Anal. 21*, 33–59.

Lebarbier, E. (2002). *Quelques approches pour la détection de ruptures à horizon fini*. Ph. D. thesis, Université Paris XI.

Lebarbier, E. (2003, February). Detecting multiple change-points in the mean of gaussian process by model selection. Technical Report 4740, INRIA.

Mallows, C. (1974). Some comments on Cp. *Technometrics 15*, 661–675.

Miao, B. Q. and L. C. Zhao (1993). On detection of change points when the number is unknown. *Chinese J. Appl. Probab. Statist. 9*(2), 138–145.

Schwarz, G. (1978). Estimating the dimension of a model. *Ann. Stat. 6*, 461–464.

Yao, Y. (1988). Estimating the number of change-points via Schwarz criterion. *Stat. & Probab. Lett. 6*, 181–189.